A COMPUTER SIMULATION FOR THE EVALUATION OF SURFACE-TO-AIR MISSILE SYSTEMS IN A CLEAR ENVIRONMENT.

Alvin F. Andrus

TR/RP No. 67

UNITED STATES
NAVAL POSTGRADUATE SCHOOL

A COMPUTER SIMULATION FOR THE EVALUATION OF SURFACE-TO-AIR MISSILE SYSTEMS IN A CLEAR ENVIRONMENT

by

Alvin F. Andrus

June 1966
TECHNICAL REPORT/RESEARCH PAPER NO. 67

UNITED STATES NAVAL POSTGRADUATE SCHOOL
Monterey, California

ABSTRACT:

A probabilistic event store computer simulation of the interactions between surface-to-air missile systems and aircraft in a non-jamming environment and over flat terrain is presented. The purpose of the model is to test the general disposition of missile areas and the associated missile system reaction times against an aircraft attack. A complete description of the model with the flow charts and CDC-FORTRAN-60 program listing is included.

This task was sponsored by: Chief of Naval Operations (OP-06C)

Prepared by: A. F. Andrus

Approved by:

J. R. Borsting

Chairman, Department of Operations Analysis

Released by:

C. E. Menneken

Dean of Research Administration

U. S. Naval Postgraduate School Technical Report/Research Paper No. 67
June 1966

UNCLASSIFIED

TABLE OF CONTENTS

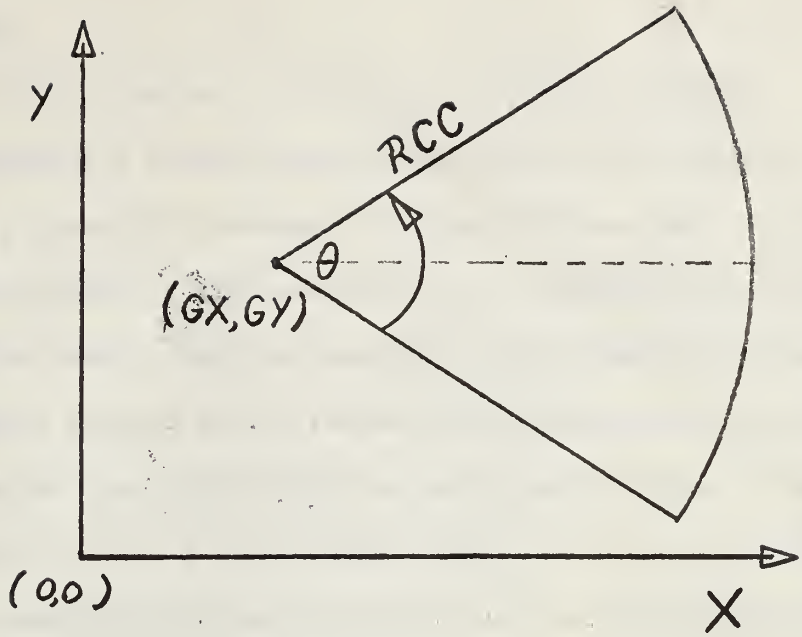
LIST OF ILLUSTRATIONS

1. INTRODUCTION

The model presented in this paper is an event store computer simulation of the interactions between surface-to-air missile systems and aircraft in a non-jamming environment and over flat terrain. The purpose of the model is to test the general disposition of missile areas and the associated missile system reaction times against an aircraft attack. The model is a probabilistic simulation in that the success or failure of any probabilistic event is determined in the model by comparing the numerical value assigned to the probability of success or failure to a program generated random number. It is the case in this model that all numbers so used are uniformly distributed. A complete description of the model is included in the following sections.

2. PLAYING AREA

The playing area for the model is that portion of a circle whose center, radius and central angle are inputs. The structure of the computer program is such that the playing area must be located entirely within the first quadrant of the X,Y plane. The numerical restrictions within the computer program are such that the central angle and radius must be less than 180 degrees and 1000 miles respectively. The playing area must also be oriented such that the bisector of the central angle is parallel to the X-axis while the central angle opens in the positive X-direction. The playing area is illustrated in Figure 1.



(GX, GY)	Coordinates of Center
RCC	Radius of Playing Area
θ	Central Angle

Note:

$$0 < GX < GX + RCC \cdot \cos \theta/2$$
$$0 < GY - RCC \cdot \sin \theta/2 \leq GY$$

$$RCC < 1000 \text{ miles}$$
$$\theta < 180 \text{ degrees}$$

Playing Area
Figure 1

3. OFFENSE

The offense consists of as many as twenty aircraft. These aircraft fly through the playing area in an attempt to penetrate the missile defenses. The entry points into the playing area for the aircraft are generated by the computer program by assuming the entry points are uniformly distributed along the arc of the circle defined by the radius of the playing area and the central angle. The flight path for each aircraft after it enters the playing area is to fly straight toward the center, (GX,GY).

The spacing time between aircraft and the speeds and altitudes of aircraft are also generated by the computer program. These values are assumed by the model to be uniformly distributed between their respective minimum and maximum values. These minimum and maximum values are inputs to the model.

The aircraft in the model play a passive role and serve only as the set of stimuli needed to cause the missile systems to act. These aircraft do not defend themselves against missile attack nor do they attack the missile areas. Under the assumption that the center of the playing area represents the bomb release line of the aircraft for their respective targets, an offensive wave of aircraft is considered successful if at least one aircraft reaches the center.

4. DEFENSE

The defense consists of as many as three missile areas with their associated missile systems. These missile areas may be located anywhere in the first quadrant. They need not be located within the playing area; however, since only the results of treating [illegible]

[illegible]

interactions occurring within the playing area are considered in the model, the sphere of influence of the missile area must include some portion of the playing area in order for the missile areas to exert any effect on the simulation results.

Associated with each missile area are the parameters needed to describe its missile system. The values of these parameters are inputs to the model and the parameters are:

(1) Search radar maximum range

(2) Missile maximum range

(3) Missile average speed

(4) The number of tracking radars

(5) The number of missile launchers

(6) Maximum and minimum time required to reload a launcher

(7) Maximum and minimum time required to acquire a target on the tracking radar

(8) Maximum and minimum time required to assess a target after missile intercept

(9) Missile single-salvo kill probability

It is an assumption of the model that all aircraft are observed by all missile areas subject to the aircraft radar horizon and the missile area search radar maximum range. It is also the case that in order to fire a missile, or salvo, at an aircraft:

(1) The aircraft must be observed at the time of fire.

(2) A missile launcher must be loaded.

(3) A tracking radar must be free in order to be used for full course missile guidance.

(4) The intercept point must be within the missile maximum range circle.

(5) The aircraft must not be past the point of closest approach to the missile area at the time of fire.

The model does not include altitude or minimum range restrictions on the missile.

The significant time delays inherent to the missile systems and included in the model are seen to be:

(1) Reload time: The amount of time required to reload a missile launcher.

(2) Acquisition time: The amount of time required, once an aircraft is observed on the search radar, to transfer the aircraft as a target to an available tracking radar.

(3) Assessment time: The amount of time the tracking radar must remain trained on the target after missile intercept in order for the results of the intercept to be observed.

All of these times when used in the model are assumed to be uniformly distributed between their maximum and minimum values.

The firing doctrine for a missile system is shoot-look-shoot at all available aircraft. That is, when a missile area has launched a salvo against a target no new salvos against that target will be launched from that missile area until that salvo has

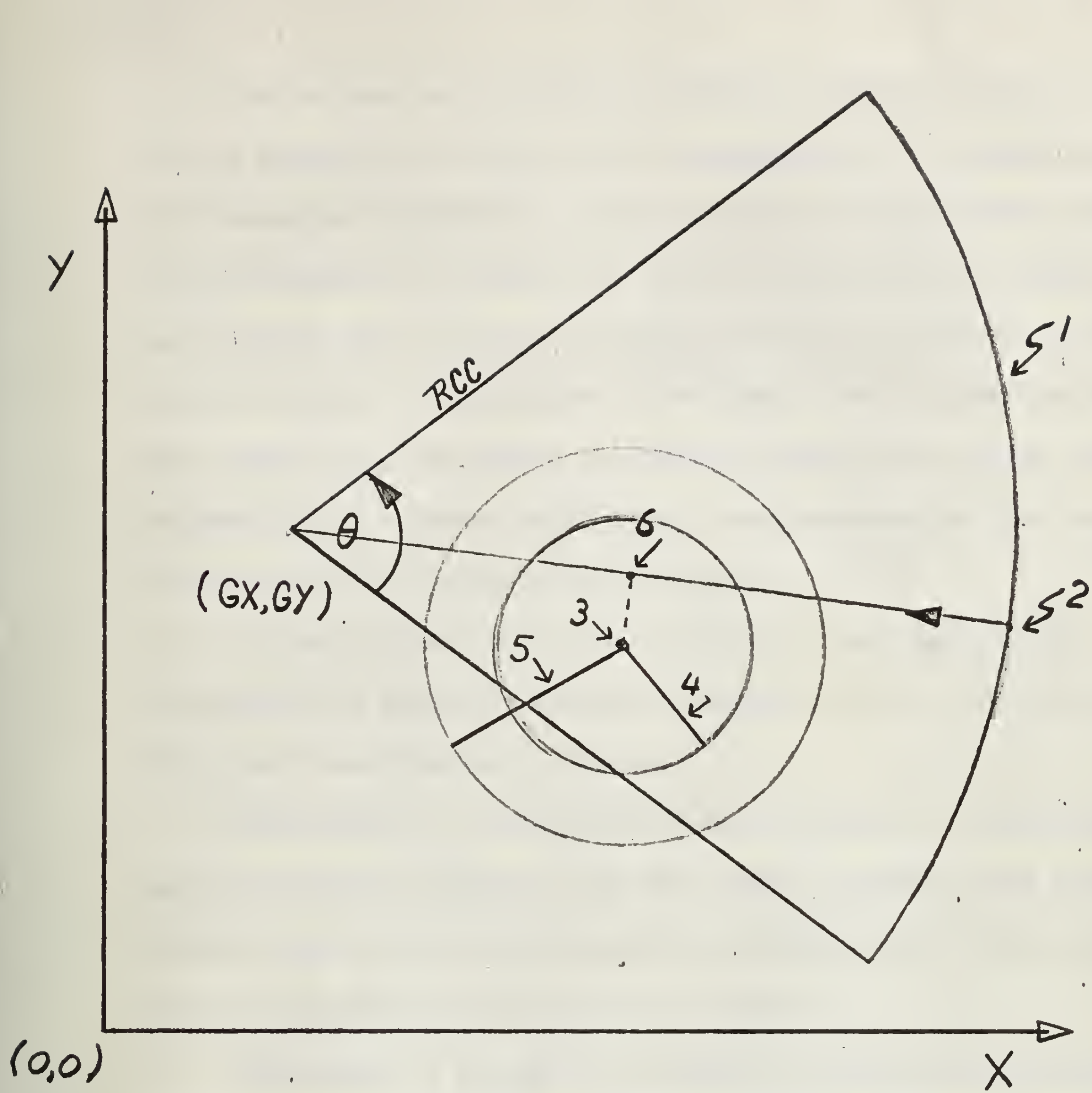
intercepted the target and the results of the intercept have been assessed. The aircraft are selected as targets, within the missile launcher and tracking radar numerical restrictions, on a first-come first-served basis.

Many of the built in characteristics of the missile system can be overridden by a proper selection of the input parameter values. For instance, by selecting a very high value, say 99, for the number of tracking radars at a missile area the availability of tracking radars will not effect the simulation results, i.e., the missile would be considered as a non-beam rider. This example is based upon the firing doctrine of the missile area and the number of aircraft in the model.

An illustration of the playing area with a typical missile area and aircraft flight path is included as Figure 2.

5. GAME DOCTRINE

With the input parameter values assigned the model considers the interactions that occur in the playing area between the missile systems and aircraft. For the given set of defensive and offensive parameters the required number of aircraft will enter the playing area at points, times, speeds and altitudes generated by the computer program. This set of aircraft will then proceed directly toward the center, (GX,GY), passing through the missile defenses.



1 Entry arc for aircraft
2 Typical aircraft entry point and flight path
3 Location of missile area
4 Missile maximum range
5 Search radar maximum range
6 Point of closest approach

Playing Area With Missile Area And Aircraft Flight Path
Figure 2

One complete pass through the computer simulation with one set of aircraft is referred to as a replication. To generate data for statistical purposes, at the completion of a replication the computer program will generate a new set of aircraft and using the same set of input values will produce another replication. The desired number of replications is an input value and must be less than twenty-one. An entire set of replications for a given number of aircraft is referred to as a run. The model output then consists of any of the following forms of output:

(1) Battle History: An event history of each replication containing the generated events of the battle in the order in which the events occur and are generated.

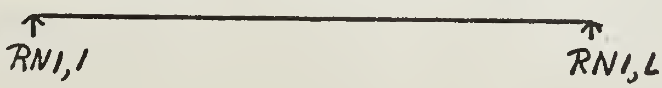
(2) Standard: A compilation of each replication containing all aircraft initial conditions and the number of salvos fired by each missile area at each aircraft and the identification of the missile area responsible for killing each aircraft.

(3) Summary: A summary of information, by totals with respect to replication, for each run including the sample mean, variance and standard deviation of all totals presented.

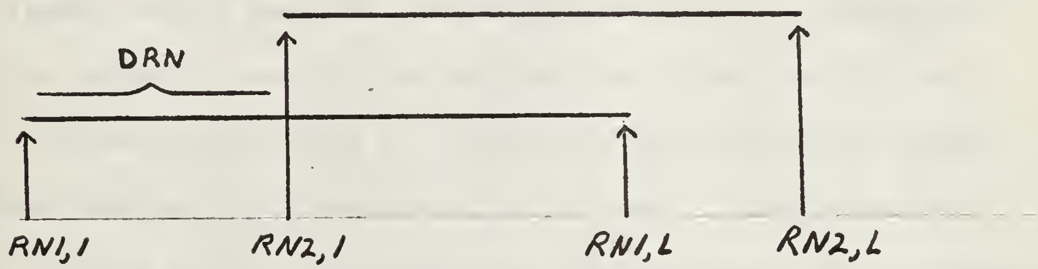
The computer program will then make as many runs as desired by the model user with an increased number of aircraft for each run. The number of aircraft in the first run, the increment for the number of aircraft in each new run, and the number of runs are input values. Of interest is that each new run is considered by

the model to be an extension of the previous run, that is, if run three contained seven aircraft and run four is to contain nine aircraft, then for all replications in run four the first seven aircraft will have entry points, altitudes, speeds and times identical to those replications in run three, etc. It is also the case that the random numbers used in the replications of a run in order to determine the outcome of probabilistic events will be used again in the replications of a new run, that is, the starting point in the sequence of random numbers used for replication two of run three will be the same as the starting point in the sequence for replication two of run four. In this manner it is hoped that any changes in the results between runs three and four can be attributed to the increase in the number of aircraft rather than to the deviations of the sets of random numbers used.

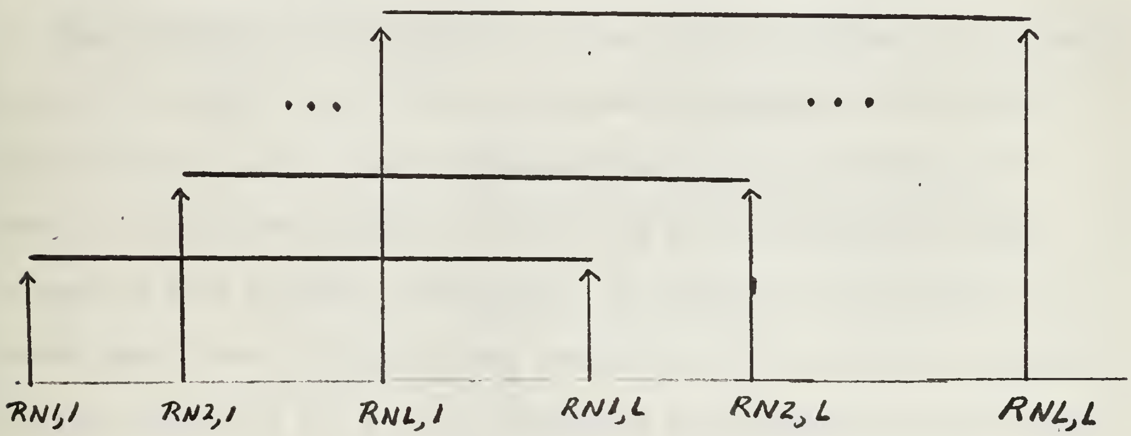
The sequences of random numbers used can also be controlled by the model user in order to reduce to a minimum the effect of gross random number fluctuations on the simulation results. The controlling input values are the initial starting point in the random number sequence for replication one of run one and an increment used to determine the starting points of all succeeding replications. These values can be selected such that the random number sequence used in a replication will be generated so that it overlaps the sequence used in the previous replication. This is illustrated as follows. If



represents the sequence of random numbers used in replication one of run one where RN1,1 and RN1,L are the first and last numbers used then the sequence for replication two can be made to overlap this sequence by choosing an increment, DRN, such that



This overlap will then continue throughout the replications as illustrated:



The numerical value assigned to DRN is not the particular value of any random number but the number of random numbers between RN1,1 and RN2,1.

When a new run is made the computer program random number generator is reset, as indicated in the previous paragraph, to begin the sequence for replication one of the new run at RN1,1. The same value of DRN is again used and in this way the sequences of random numbers used for the new replications will correspond to the sequences used for the replications of the previous run.

The computer program is constructed such that either or both of two missile firing procedures may be used. These procedures are referred to as uncoordinated and coordinated and the procedure used in the simulation is determined by the model user as an input to the model. The uncoordinated missile firing procedure allows all missile areas in the simulation to fire missiles at all aircraft that can possibly be fired upon while the coordinated missile firing procedure allows a missile area to fire missiles at an aircraft only if no other missile area is currently engaging that aircraft. For any given set of input parameters the computer program will make the necessary replications and runs of the simulation using either or both of these procedures. Of interest is that when the model user elects to employ both procedures, they are not intermixed in the simulation but are run separately and simulation results are presented so that comparisons can be made as a function of the procedure used. It is also the case that when both missile firing procedures are used the same sets of aircraft and sequences of random numbers are used in the corresponding replications and runs

[illegible]

of the simulation. This is illustrated in Figure 3 which displays the overall simulation structure.

6. EVENTS

As mentioned earlier the model is an event store computer simulation, i.e., all actions that are to occur in the simulation are dynamically generated by the computer program as a result of previous simulation actions and are listed chronologically in an Event Store List. Each of the actions included in the simulation assumes the form of a computer program subroutine, called an event, and the information pertaining to the action on the Event Store List is the information needed to execute the proper subroutine. There are only four major actions included in the simulation as events and these events are:

(1) Fire Missile Salvo

(2) Missile Intercept

(3) Reload Missile Launcher

(4) Free the Tracking Radar from an Intercepted Target

Each of the computer program subroutines representing these events uses as input parameters the following information:

(1) Time event is to occur

(2) Identification of Event

(3) Identification of Aircraft

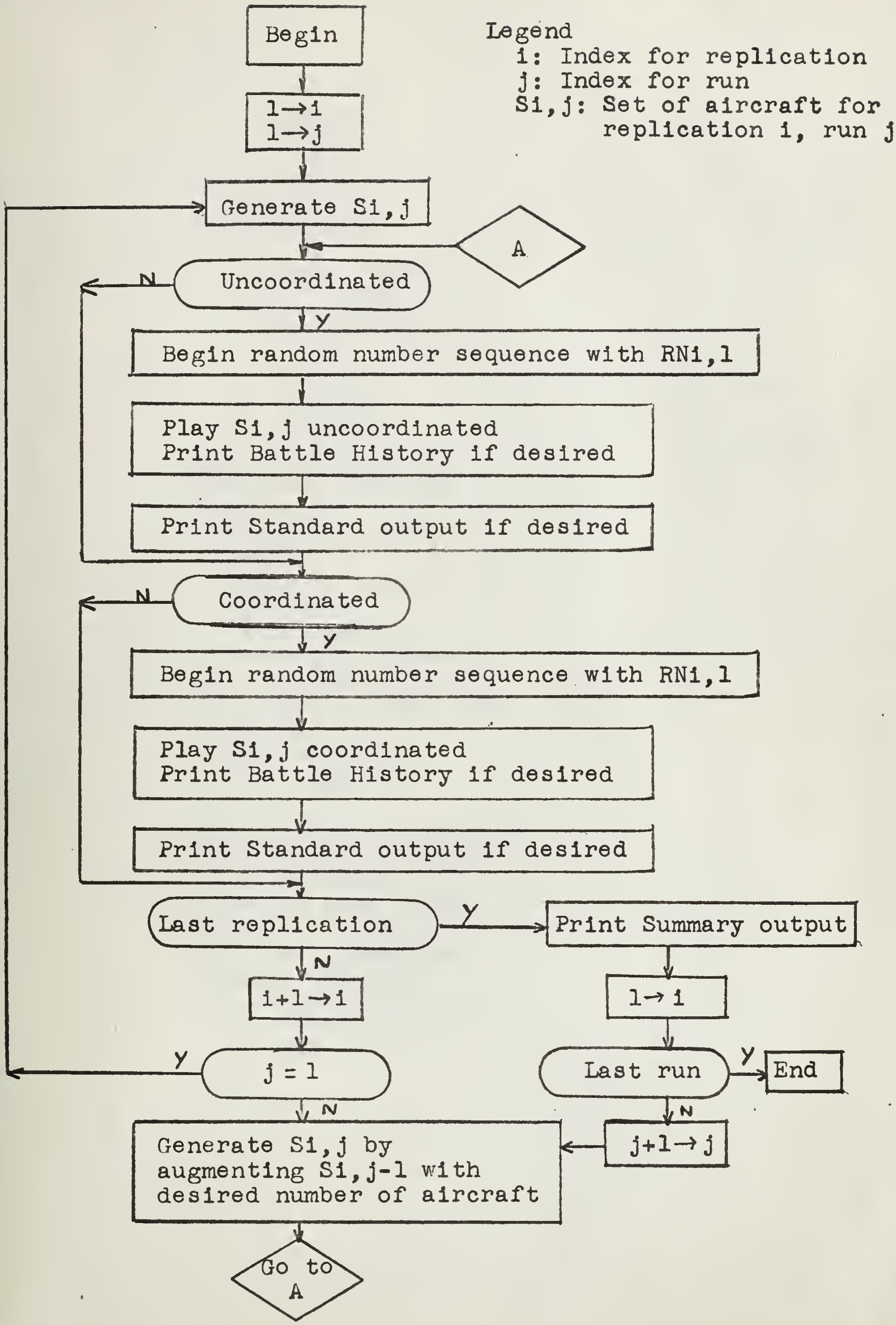
(4) Identification of Missile Area

A fifth event, Change Engagement Status, is also part of the simulation but does not represent an interaction between aircraft

and missile areas. The Change Engagement Status subroutine provides the necessary program bookkeeping for logical consistency when the coordinated missile firing procedure is used.

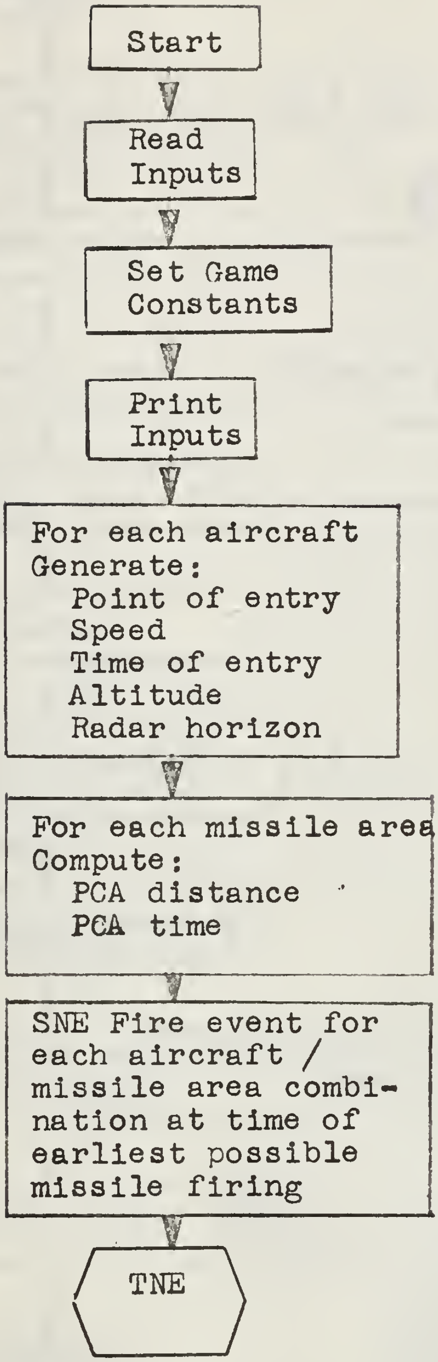
The dynamic process of simulating one air battle from start to finish forms the executive routine for the computer simulation. This executive routine consists of nothing more than two program subroutines referred to as SNE and TNE. SNE, Store Next Event, is the subroutine that takes the generated information pertaining to an interaction and properly places this information on the Event Store List. TNE, Take Next Event, is the subroutine that, at the completion of any of the five events, interrogates the information on the Event Store List and transfers control of the computer program to the proper subroutine.

General flow charts describing the logic included in each event of the simulation plus the interrelationship of events is included as Figure 4 through Figure 9.

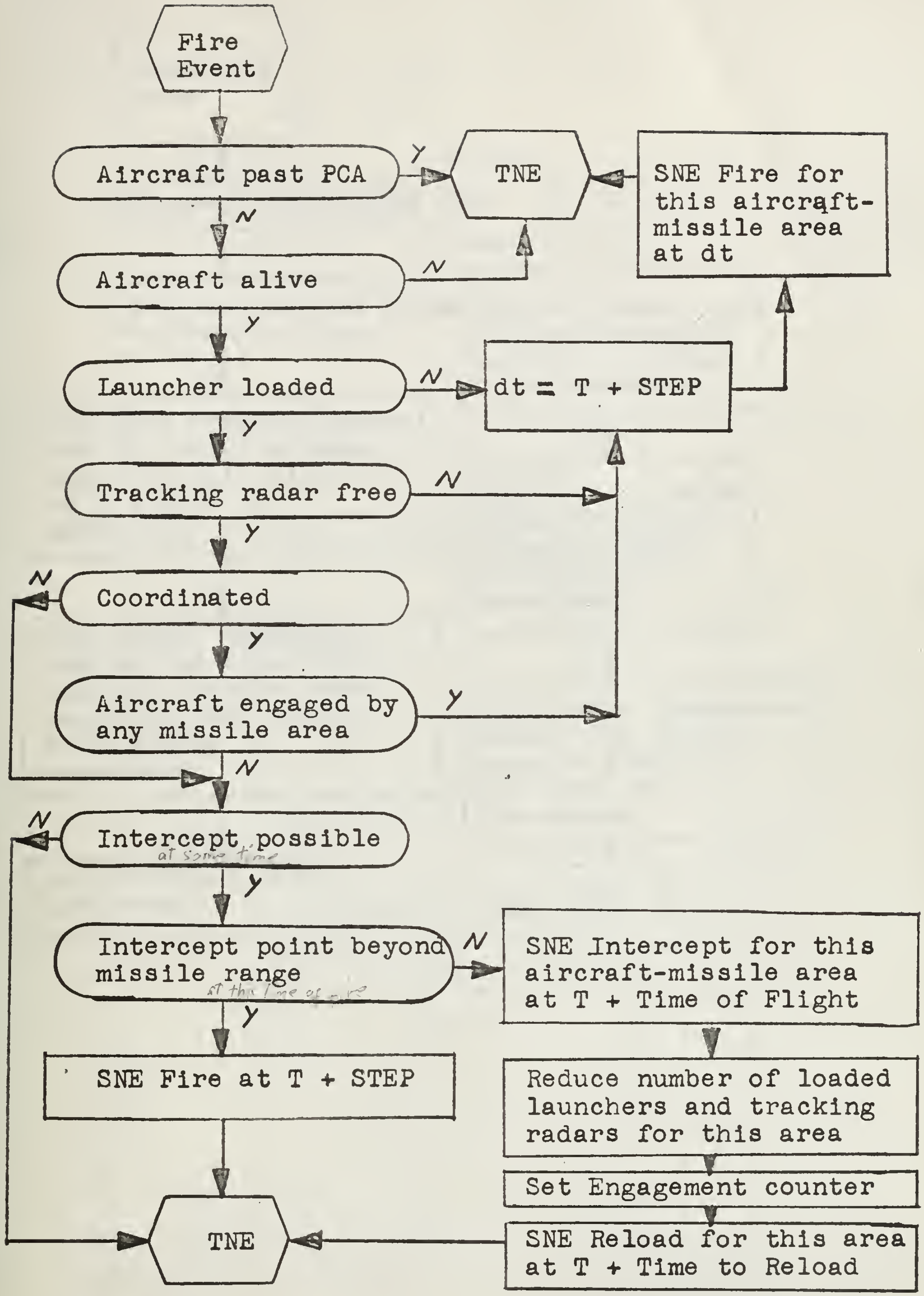


Overall Simulation Structure

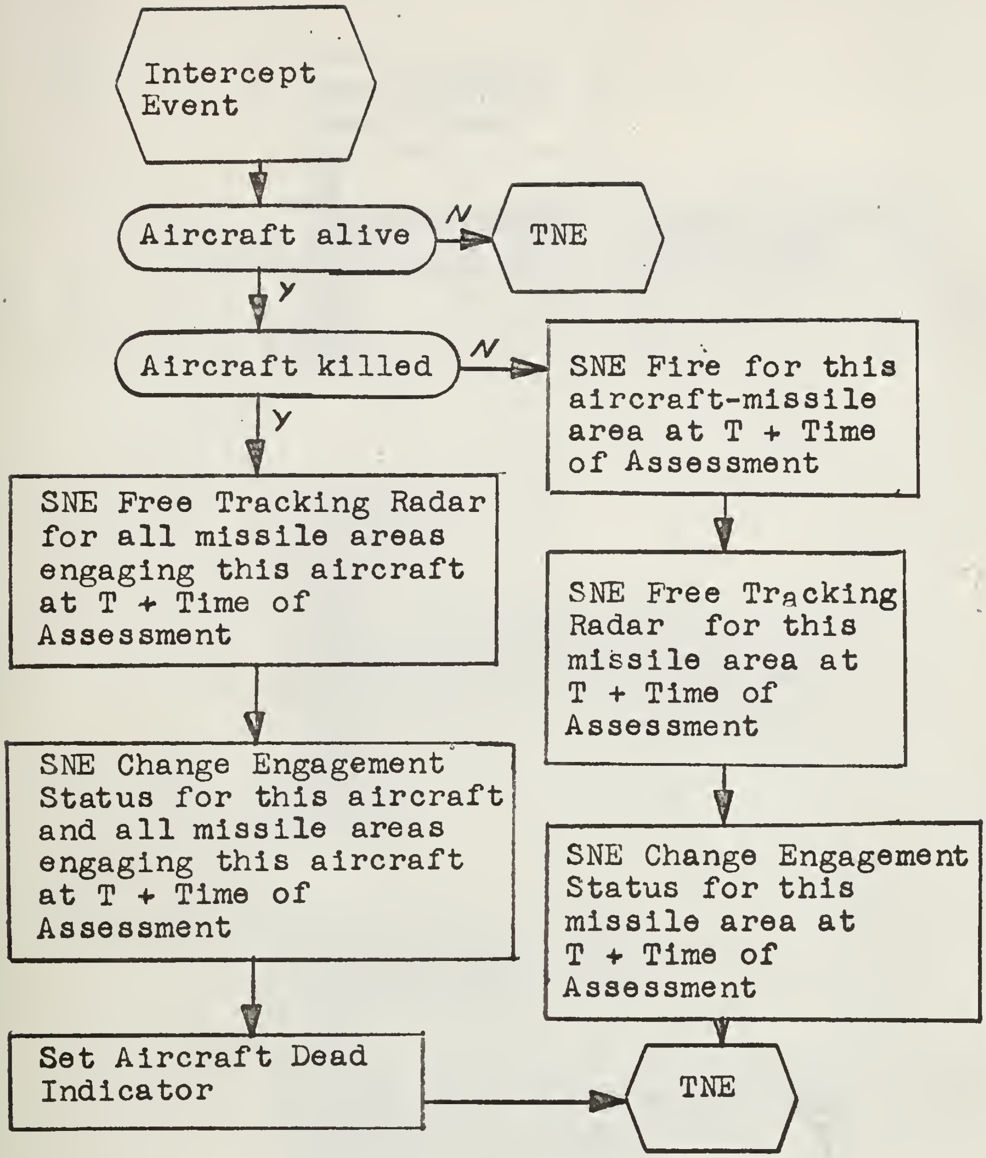
FIGURE 3



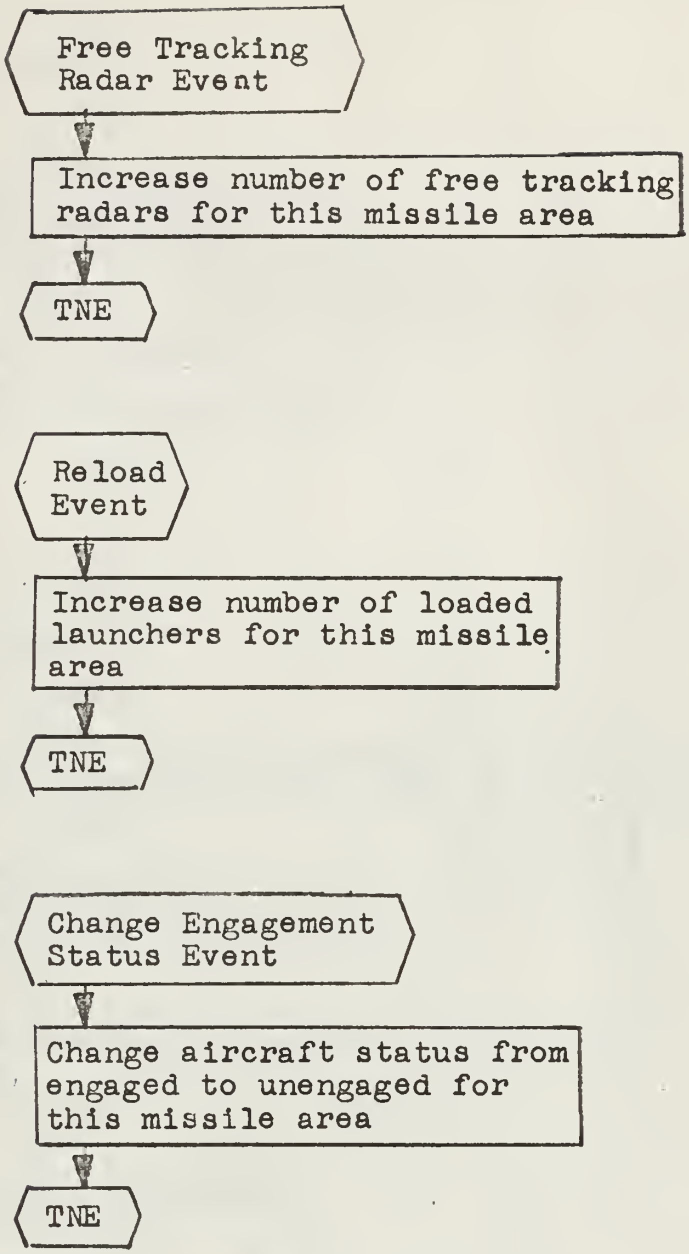
Begin Simulation Logic
FIGURE 4



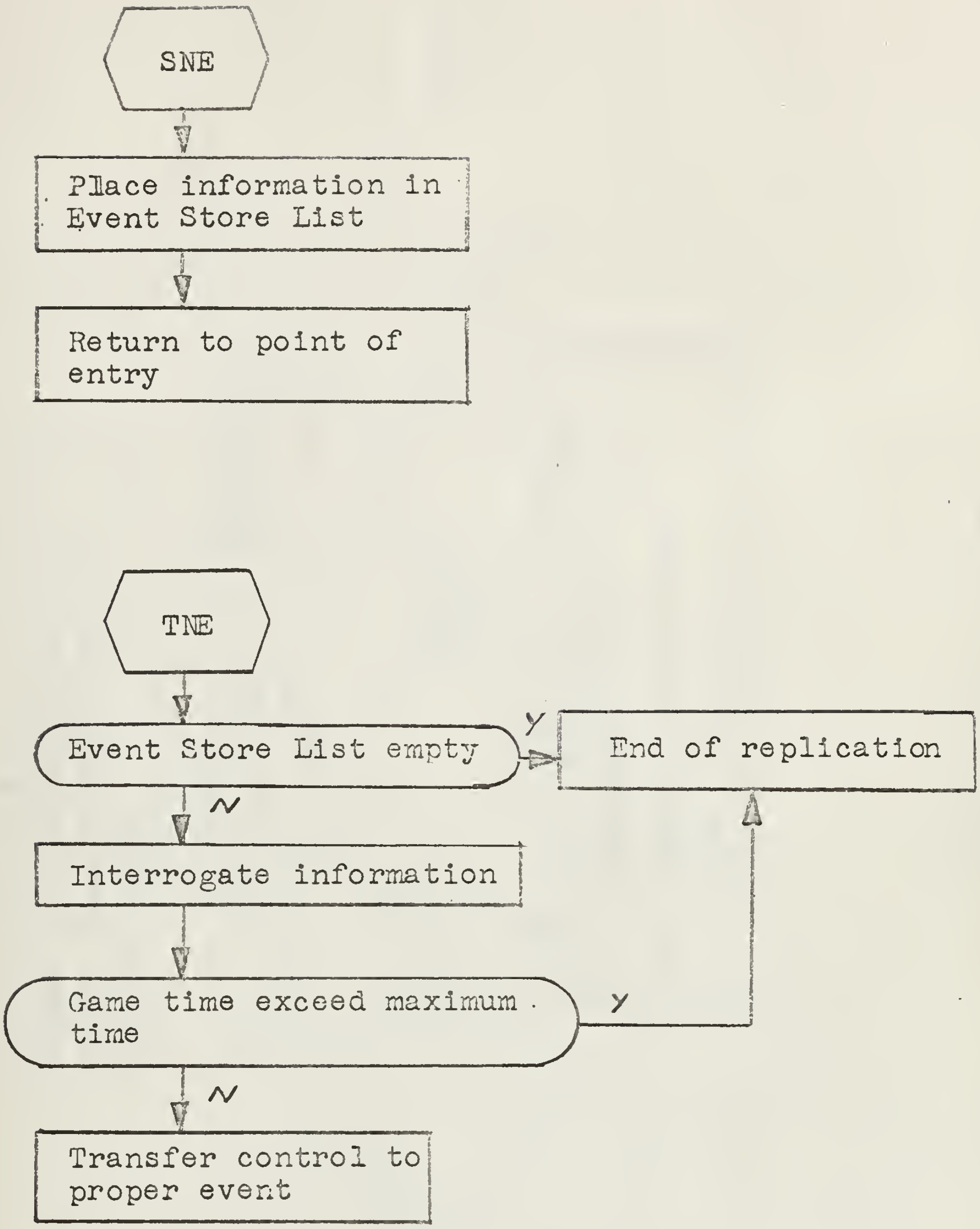
Fire Event Logic
FIGURE 5



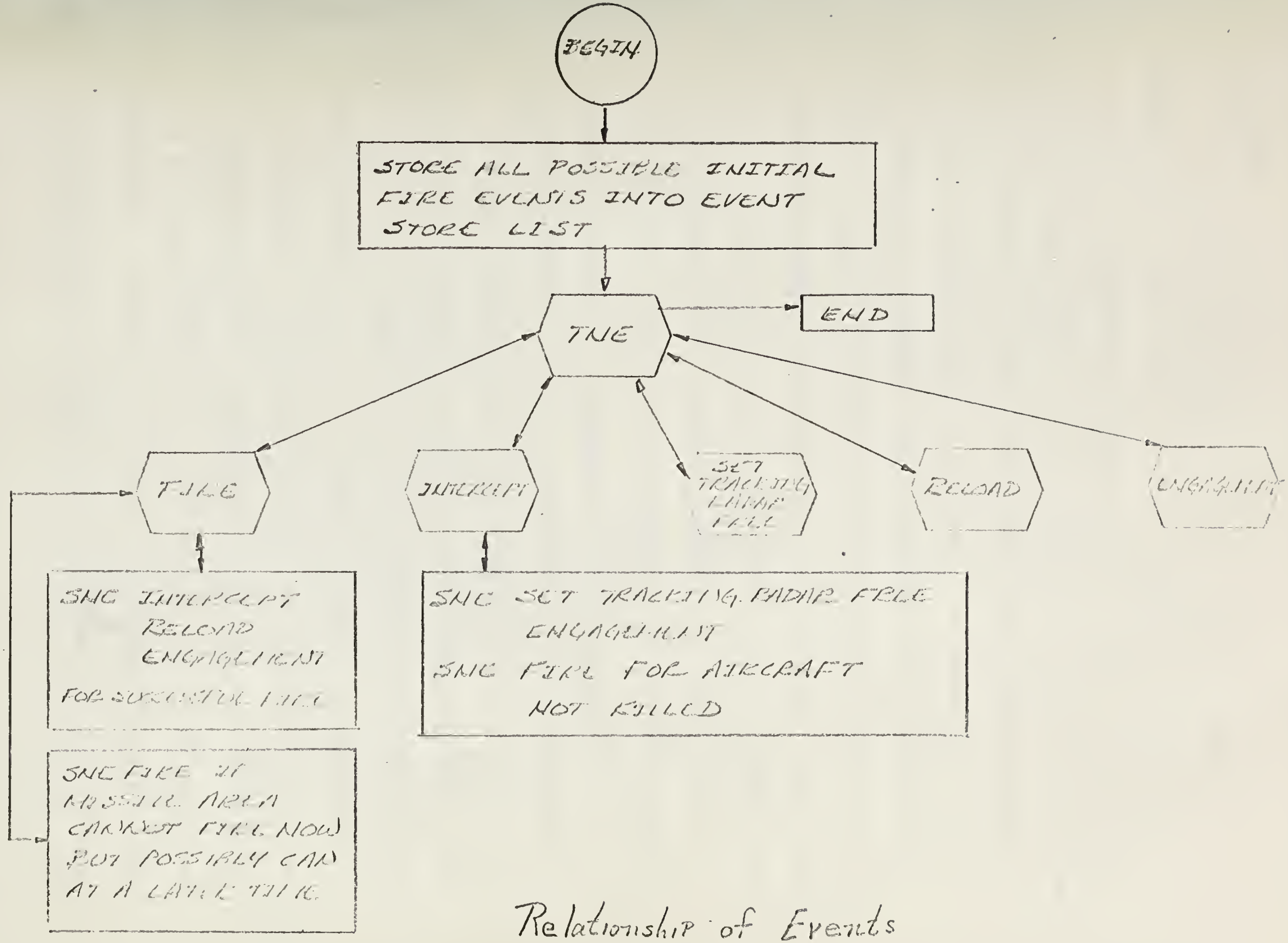
Intercept Event Logic
FIGURE 6



Free Tracking Radar Event Logic
Reload Event Logic
Change Engagement Status Event Logic
FIGURE 7



SNE and TNE Logic
FIGURE 8



Relationship of Events
Figure 9

7. INPUT

All input to the computer program is contained on four punched cards. The format for this information and the units to be used are detailed in Figures 10, 11, and 12. With one set of input cards a model user can simulate the interactions involved with a fixed set of missile areas, with fixed values describing the missile systems, and sets of aircraft of increasing numbers. The simulation is repeated using different sets of aircraft and different sequences of random numbers in order to provide data for statistical analysis. In order for the model user to vary parameter values other than the number of aircraft a new set of input cards must be supplied to the computer program. With the proper input values the model can be made deterministic and used for straight forward calculations of sequences of intercept points, etc. A complete description of each input variable follows Figure 12.

For input variables whose names begin with I, J, K, L, M, or N the input values must be integers in the units specified. For all other input variables the input values are floating point and decimal points can be either omitted or used as desired in the space allotted. For example the input variable name AMAX representing the maximum altitude of aircraft is a floating point value whose units are thousands of feet. In the three columns allotted for AMAX on Card 3 the model user could enter Blank-5-0 or 5-0-. for fifty thousand feet and could also enter Blank-.-5 for five hundred feet.

COLUMNS	NAME	RANGE	UNITS
1	NMISAR	≤ 3	
2-3	MTRF(1)	< 100	
4-5	MTRF(2)	< 100	
6-7	MTRF(3)	< 100	
8	MAL(1)	< 10	
9	MAL(2)	< 10	
10	MAL(3)	< 10	
11-13	AX(1)	< 1000	MILES
14-16	AX(2)	< 1000	MILES
17-19	AX(3)	< 1000	MILES
20-22	AY(1)	< 1000	MILES
23-25	AY(2)	< 1000	MILES
26-28	AY(3)	< 1000	MILES
29-31	RMR(1)	< 1000	MILES
32-34	RMR(2)	< 1000	MILES
35-37	RMR(3)	< 1000	MILES
38-40	RMAX(1)	< 1000	MILES
41-43	RMAX(2)	< 1000	MILES
44-46	RMAX(3)	< 1000	MILES
47-50	AVS(1)	< 10000	MILES/HR
51-54	AVS(2)	< 10000	MILES/HR
55-58	AVS(3)	< 10000	MILES/HR
59	IEND	1 OR 0	
60-72	NAME OF PROGRAM USER		

ALL INPUT INFORMATION IS RIGHT-JUSTIFIED IN THE SPACE ALLOTTED.

FORMAT FOR INPUT DECK CARD 1

FIGURE 10

COLUMNS	NAME	RANGE	UNITS
1-2	PK(1)	< 100	%
3-4	PK(2)	< 100	%
5-6	PK(3)	< 100	%
7-9	ATM(1)	< 1000	MINUTES
10-12	ATM(2)	< 1000	MINUTES
13-15	ATM(3)	< 1000	MINUTES
16-18	ATX(1)	< 1000	MINUTES
19-21	ATX(2)	< 1000	MINUTES
22-24	ATX(3)	< 1000	MINUTES
25-27	ASM(1)	< 1000	MINUTES
28-30	ASM(2)	< 1000	MINUTES
31-33	ASM(3)	< 1000	MINUTES
34-36	ASX(1)	< 1000	MINUTES
37-39	ASX(2)	< 1000	MINUTES
40-42	ASX(3)	< 1000	MINUTES
43-45	RTM(1)	< 1000	MINUTES
46-48	RTM(2)	< 1000	MINUTES
49-51	RTM(3)	< 1000	MINUTES
52-54	RTX(1)	< 1000	MINUTES
55-57	RTX(2)	< 1000	MINUTES
58-60	RTX(3)	< 1000	MINUTES

ALL INPUT INFORMATION IS RIGHT-JUSTIFIED IN THE SPACE ALLOTTED.

FORMAT FOR INPUT CARD 2
FIGURE 11

COLUMNS	NAME	RANGE	UNITS
1-3	NBMAX	≤ 20	
4-7	BSX	< 10000	MILES/HR
8-11	BSM	< 10000	MILES/HR
12-14	BTX	< 1000	MINUTES
15-17	BTM	< 1000	MINUTES
18-20	NBOMB	≤ 20	
21-23	NBDEL	≤ 20	
24-26	GX	< 1000	MILES
27-29	GY	< 1000	MILES
30-32	RCC	< 1000	MILES
33-35	CPA	< 180	DEGREES
36-37	TMAX	< 100	HOURS
38	OUTPUT	1 OR 0	
39-41	NR1	< 1000	
42-44	NRDEL1	< 1000	
45-47	NR2	< 1000	
48-50	NRDEL2	< 1000	
51-52	NRPL	≤ 20	
53-55	AMAX	< 100	K FEET
56-58	AMIN	< 100	K FEET

CARD 4

COLUMNS	NAME	RANGE	UNITS
1-3	STEP		MINUTES
4-5	NFLAG	0,1 OR 2	
6-7	IHIST	0 OR 1	

ALL INPUT INFORMATION IS RIGHT-ADJUSTED IN THE SPACE ALLOTTED.

FORMAT FOR INPUT DECK CARD 3 AND 4

FIGURE 12

DESCRIPTION OF INPUTS

CARD 1:

NMISAR	Number of missile areas in the game simulation.
MTRF(I)	Number of tracking radars assigned to missile area I.
MAL(I)	Number of missile launchers assigned to missile area I. The model assumes one missile launcher represents the salvo size. The salvo size is not explicitly included as an input value.
AX(I)	X-coordinate of missile area (I).
AY(I)	Y-coordinate of missile area (I).
RMR(I)	Search Radar Maximum Range of missile area (I).
RMAX(I)	Missile Maximum Range for missile area (I).
AVS(I)	Average Speed of missile for missile area (I).
IEND	To change the values of any input parameters, except the number of aircraft, another set of 4 input cards is required. The new set follows the original set of 4 cards in the data deck ad-infinitum. IEND is a program flag to indicate the last set of data. 0 ⇒ Last set of data 1 ⇒ Another set of 4 input cards to follow
NAME	Programmer name or identification. Maximum of 13 Hollerith characters.

a

Note: If the number of missile areas used is less than three the computer program assumes the missile areas are identified in sequence beginning with missile area 1. For example, if only two missile areas are being used the input data must correspond to missile areas 1 and 2.

[illegible]

[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]

[illegible]

[illegible]	[illegible]

[illegible]

CARD 2:

PK(I) Missile salvo Kill Probability for missile area (I). Salvo size is not explicitly included in the model. Salvo size is included by coordinating the number of missile launchers, MAL(I), with PK(I).

ATM(I) Minimum Acquisition Time for missile area (I). ?

ATX(I) Maximum Acquisition Time for missile area (I).

ASM(I) Minimum Assessment Time for missile area (I).

ASX(I) Maximum Assessment Time for missile area (I).

RTM(I) MInimum Reload Time for missile area (I).

RTX(I) Maximum Reload Time for missile area (I).

CARD 3:

NBMAX	Maximum Number of Aircraft in the last run.
BSX	Maximum Aircraft Speed.
BSM	Minimum Aircraft Speed. For all aircraft to enter at the same speed BSX and BSM should be the same value.
BTX	Maximum Time Spacing between aircraft.
BTM	Minimum Time Spacing between aircraft. For all aircraft to enter at the same time BTX and BTM should be set to zero. For all aircraft to enter at a definitely fixed time spacing between aircraft BTX and BTM should be the same value.
NBOMB	Number of Aircraft in the first run.
NBDEL	The increment used to increase the number of aircraft for a new run. The number of runs to be made with one set of input cards will be the minimum integer value of N such that

$$NBOMB + (N)(NBDEL) \geq NBMAX$$

GX	The X-coordinate of the center.
GY	The Y-coordinate of the center.
RCC	The length of the Radius of the playing area.
CPA	The Central Playing Angle defining the playing area and aircraft line of entry.
TMAX	Maximum Game Time. The simulation will automatically end when no more interactions are possible. If however the model user would only prefer results for a fixed period of time, that time can be entered here. If complete results are desired, use a very large number such as 99. See note.
OUTPUT	Program flag used to indicate if Standard and/or Summary output is desired.

0: Standard and Summary output
1: Summary only

Note: The first aircraft enters the playing area at time = 1 hour.

[illegible]

CARD 3: (Continued)

OUTPUT (Continued)	Normally only Summary output is desired. The use of Standard output greatly increases the computer running time.
NR1	The position of RN1,1, the first random number selected for the simulation. Since this position is reached by generating and discarding the numbers in previous positions use a small value, say 200, to save computer time.
NRDEL1	The increment for the sequential jump on NR1 for a new replication. If NR1 and NRDEL1 are the same on different sets of input cards, the same sequences of random numbers will be used for the simulations represented by the different sets of input cards.
NR2	The position of the first Random Number used for the generation of aircraft entry points, times, speeds and altitudes.
NRDEL2	The increment for the sequential jump on NR2 for a new replication. If NR2 and NRDEL2 are the same on different sets of input cards, the same sequences of random numbers will be used to generate the sets of aircraft for the simulations represented by the different sets of input cards. If it is the case that all other aircraft and playing area information is the same, then the sets of aircraft used in the simulations will be exactly the same.
NRPL	The number of Replications to be played.
AMAX	Maximum Aircraft Altitude.
AMIN	Minimum Aircraft Altitude. Aircraft altitude is used only to compute the radar horizon using the expression $RH = 1.25\sqrt{A}$ where RH is Radar Horizon in miles and A is Aircraft altitude in feet. An altitude of zero yields a radar horizon of zero and consequently for aircraft with an altitude of zero no missile area interactions will occur. For all aircraft to have the same altitude AMAX and AMIN should be the same value.

CARD 4:

STEP — Step is a time step value used for computer efficiency. When a missile salvo can not be fired at an aircraft because of (a) no launcher is loaded, (b) no tracking radar is free, or (c) if the missile area fired now the computed intercept point would be beyond missile maximum range, then a new Fire Event is stored at current time plus the value of Step. This value must not be zero. For normal aircraft speeds a decent value of Step is probably one minute.

NFLAG — This value controls the missile firing procedures:

0: Both uncoordinated and coordinated procedures are used in the simulation.
1: Uncoordinated procedure used only
2: Coordinated procedure used only

See the previous section on DOCTRINE for a detailed discussion. If in the play of the simulation only one missile area or one aircraft is used, or if the spheres of influence of the missile areas do not overlap then the uncoordinated procedure results will be the same as the coordinated and the value of NFLAG should be 1 in order to save computer time.

IHIST — This value controls the battle history output.

0: No battle history output
1: A battle history output is printed

The use of the battle history output greatly increases computer running time and should be used judiciously if at all. Therefore this value should normally be set to zero.

8. OUTPUT

As mentioned earlier, the computer program presents three different sets of simulation results; Battle History, Standard and Summary output. The Battle History and Standard output appear for each replication and are printed only on request, i.e., when the input values for IHIST and OUTPUT are properly set. See Section 7 for the details of input. The Summary output is always printed for each run. Computer running time is greatly increased with the use of the Battle History or Standard output. This section presents the format of the output so that the model user may interpret the simulation results.

The Battle History is a list of all the events stored by SNE and executed by TNE during the replication in the order in which the events were stored and executed. The Battle History format is illustrated in Figure 13. Each line of four numbers corresponds to an event and must be decoded. The events being executed by TNE are represented by the lines beginning on the left while the indented lines represent events that are being stored as a result of the event being executed. The decoding format is as follows:

Time of Event	Type of Event	Aircraft #	Missile Area #

The time appears in hours and the aircraft and missile areas are identified by number. The type of event is decoded as follows:

[illegible]

[illegible]

[illegible]

[illegible]

NUMBER	EVENT
1	Reload
2	Set Tracking Radar Free
3	Intercept
4	Change Engagement Status
5	Fire

As an example the Battle History presented in Figure 13 could be read as follows. Fire events were stored for Aircraft 1 and missile areas 1 and 3 at 1.205 and 1.171 hours respectively and for aircraft 2 and missile areas 1 and 2 at 1.256 and 1.210 hours respectively. It must be pointed out that this is the order in which the events were stored and is not the order in which the events appear on the Event Store List. The order of the events on the Event Store List is chronological. The generation of these four events corresponds to the program logic illustrated in Figure 4. The next line is not indented and represents a TNE, i.e., at 1.171 hours missile area 3 attempted to fire at aircraft 1. As a result of this Fire event an Intercept event is stored at 1.203 hours and the launcher is scheduled to be reloaded at 1.175 hours. The execution of the Reload and Intercept events appears next since they were the two earliest events on the Event Store List. As a result of the Intercept event a Fire event is stored at 1.255 hours indicating the salvo did not kill the aircraft and Set Tracking Radar Free and Change Engagement Status events are also stored at

1.255 hours. The history of the battle can thus be read from the Battle History. Notice that at 1.255 hours missile area 3 is scheduled to fire at aircraft 1 and no events are stored as a result of the scheduled firing; this indicates that at 1.255 hours aircraft 1 was either dead, beyond the point of closest approach or the game geometry is such that a solution to the intercept equation does not exist. It can be seen that aircraft 1 was killed at 1.233 hours by missile area 1; this is indicated since no Fire event is stored as a result of the Intercept event at 1.233 hours.

The Standard output format is illustrated in Figure 14 along with an explanation of all the information presented. The example of Figure 14 can be seen to agree with the Battle History of Figure 13. The first row of this Standard output contains information pertaining to aircraft 1 and can be read as follows; in replication 5 using the uncoordinated missile firing procedure aircraft 1 entered the simulation at 1.000 hours at X = 169.47 miles and Y = 571.44 miles with a speed of 1241.62 miles/hour and at an altitude of 5.68 thousand feet; missile area 2 could not fire at aircraft 1 and the earliest time of fire from missile areas 1 and 3 was 1.205 and 1.171 hours respectively; a total of two salvos was fired at aircraft 1, one each from missile area 1 and 3 and the aircraft was killed by a salvo fired by missile area 1.

		1.205	5	1	1
		1.171	5	1	3
		1.256	5	2	1
		1.210	5	2	2
1.171	5	1	3		
		1.203	3	1	3
		1.175	1	1	3
1.175	1	1	3		
1.203	3	1	3		
		1.255	5	1	3
		1.255	2	1	3
		1.255	4	1	3
1.205	5	1	1		
		1.233	3	1	1
		1.209	1	1	1
1.209	1	1	1		
1.210	5	2	2		
		1.234	3	2	2
		1.214	1	2	2
1.214	1	2	2		
1.233	3	1	1		
		1.287	2	1	1
		1.287	4	1	1
		1.285	2	1	3
		1.285	4	1	3
1.234	3	2	2		
		1.289	2	2	2
		1.289	4	2	2
1.255	2	1	3		
1.255	4	1	3		
1.255	5	1	3		
1.256	5	2	1		
1.285	2	1	3		
1.285	4	1	3		
1.287	2	1	1		
1.287	4	1	1		
1.289	2	2	2		
1.289	4	2	2		

Battle History
Figure 13

UNCOORDINATED ATTACK RESULTS REP 5

BMR	SMA1	SMA2	SMA3	TOTAL	A/K	XCOORD	YCOORD	BRH	ALTITUDE	SPEED	TIME	TIME(1)	TIME(2)	TIME(3)
1	1	0	1	2	1	169.47	571.44	216.51	5.68	1241.62	1.000	1.205	.000	1.171
2	0	1	0	1	2	225.47	72.93	216.51	5.68	1121.85	1.043	1.256	1.210	.000

where:

BMR	The aircraft number
SMA1	The total number of salvos fired at this aircraft by missile area number 1
SMA2	Same as SMA1 for missile area number 2
SMA3	Same as SMA1 for missile area number 3
TOTAL	The total number of salvos fired at this aircraft by all missile areas
A/K	The number of the missile area killing the aircraft. A/K = 0 indicates the aircraft was not killed and penetrated the defenses.
XCOORD	The X-coordinate of aircraft entry point, in miles, into the playing area.
YCOORD	The Y-coordinate of aircraft entry point, in miles, into the playing area.
BRH	Aircraft radar horizon in miles
ALTITUDE	The aircraft altitude in thousands of feet
SPEED	The aircraft speed in miles/hour
TIME	The aircraft entry time in hours
TIME(1)	The earliest possible time, in hours, that missile area 1 can fire at the aircraft due to radar horizon, search radar maximum range and missile maximum range. A zero value indicates this aircraft can not be fired at by this missile area.
TIME(2)	Same as TIME(1) for missile area number 2
TIME(3)	Same as TIME(1) for missile area number 3

STANDARD OUTPUT FORMAT
FIGURE 14

The Summary output presented at the end of each run is illustrated in Figure 15 along with an explanation of the format used. This summary consists of the total number of aircraft kills and the total number of salvos fired by missile areas for each replication. The sample mean, variance and standard deviation of the data presented is also included. It can be seen that the last line of blocks 1 and 3 corresponds to a summary of the information contained in the examples of the Battle History and Standard output.

KILLS

NBOMB	REP	MA1	MA2	MA3	TOTAL	MEAN	VAR	SD
2	1	.	.	1.	1.	.33	.22	.47
2	2	1.	.	1.	2.	.67	.22	.47
2	3	1.	1.	.	2.	.67	.22	.47
2	4	.	1.	.	1.	.33	.22	.47
2	5	1.	1.	.	2.	.67	.22	.47

Block 1

	MA1	MA2	MA3	TOTAL
MEAN	.60	.60	.40	1.60
VAR	.24	.24	.24	.24
SD	.49	.49	.49	.49

Block 2

PROBABLITY OF SURVIVAL = .60

SHOTS FIRED

MA1	MA2	MA3	TOTAL	MEAN	VAR	SD
2.	2.	2.	6.	2.00	.00	.00
1.	.	2.	3.	1.00	.67	.82
1.	1.	1.	3.	1.00	.00	.00
1.	1.	1.	3.	1.00	.00	.00
1.	1.	1.	3.	1.00	.00	.00

Block 3

MA1	MA2	MA3	TOTAL
1.20	1.00	1.40	3.60
.16	.40	.24	1.44
.40	.63	.49	1.20

Block 4

Summary Output Format

Figure 15

where for block 1:

NBOMB	The number of aircraft entered in this replication
REP	The number of the replication
MA1	The total number of aircraft killed by missile area 1 in this replication
MA2	Same as MA1 for missile area 2
MA3	Same as MA1 for missile area 3
TOTAL	The total number of aircraft killed by all missile areas in this replication
MEAN	The sample mean of (MA1,MA2,MA3)
VAR	The sample variance of (MA1,MA2,MA3)
SD	The sample standard deviation of (MA1,MA2,MA3)

for block 2:

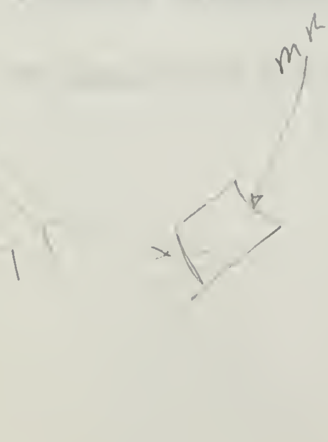
MEAN	The sample mean of the number of aircraft killed by each and by all missile areas. The entry for MEAN under MA1, 5.80, is the sample mean of the column of numbers under MA1 in block 1
VAR	The sample variance for the same set of data
SD	The sample standard deviation for the same set of data

$$\text{Probability of Survival} = \frac{\text{Number of replications in which all aircraft are killed}}{\text{Total number of replications}}$$

and represents the probability that an aircraft attack will not penetrate the missile defenses.

The same information is then presented for the total number of salvos fired, i.e., blocks 3 and 4.

SUMMARY OUTPUT FORMAT
FIGURE 15 (Continued)



9. REMARKS

No significant results have been obtained to date with the model presented. This report is the culmination of effort to build the model and make it available for use to interested parties; in particular to the students of Operations Analysis at the U. S. Naval Postgraduate School. A parametric study using the model is now underway.

The appendices to this report contain the detailed flow charts of the computer program, the CDC-FORTRAN-60 program listing and the format for using the program at the Postgraduate School.

The author thanks Miss Patricia Hoang for her able assistance in programming and debugging the computer program and Mr. Cloyd Marvin of The Johns Hopkins University Applied Physics Laboratory for the IBM-7090 FAP version of the random number generator.

APPENDIX I

Detailed Flow Charts

This appendix contains the detailed flow charts for the computer program. The flow chart pages are sequenced from A to AG and the flow chart symbols for logical continuity from page to page are:

and

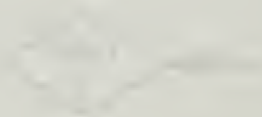
The flow chart symbol for logical continuity on the same page is:

[illegible]

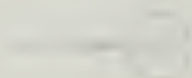
[illegible]

[illegible]

[illegible]



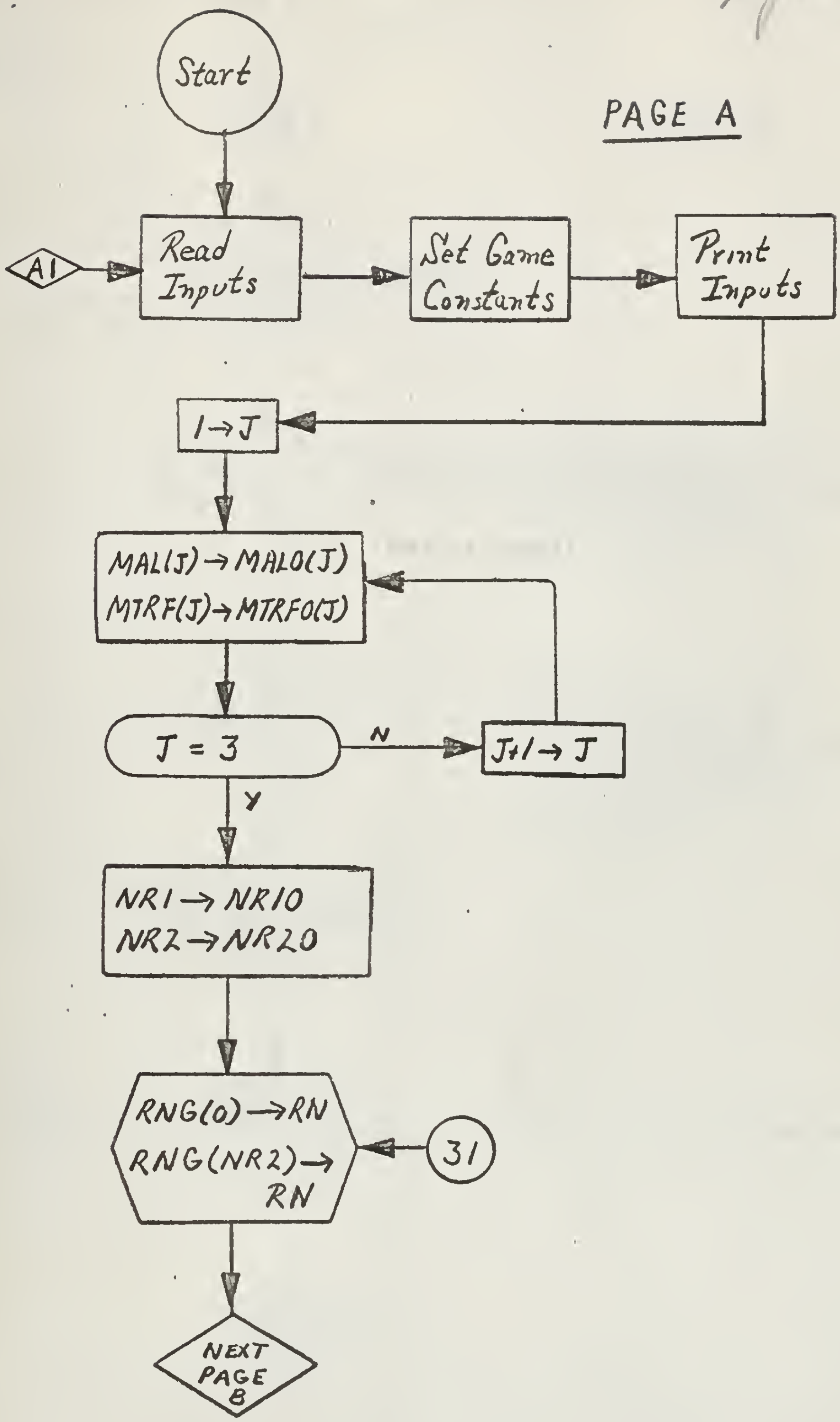
[illegible]



[illegible]

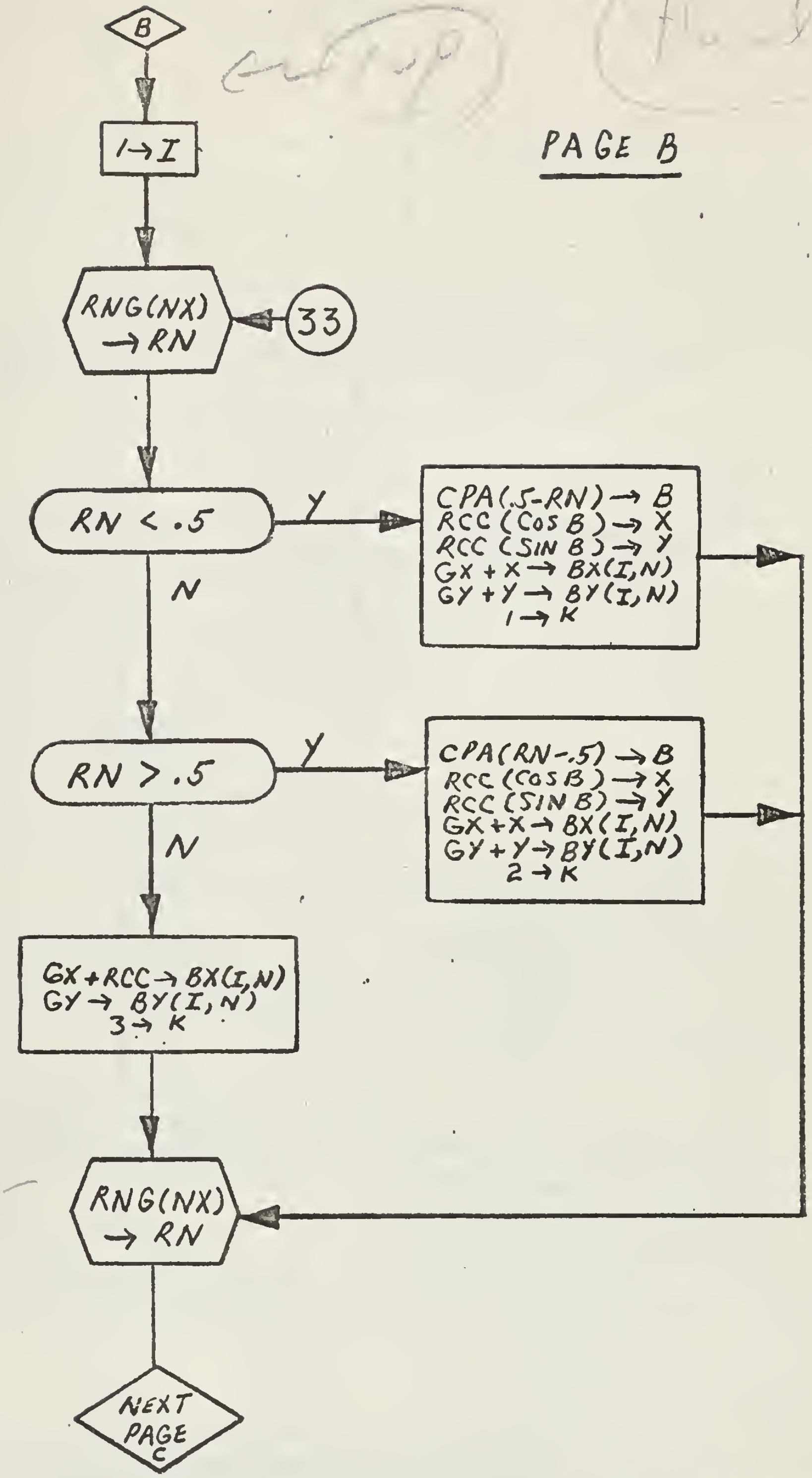
[illegible]

PAGE A


Start
A1
Read Inputs
Set Game Constants
Print Inputs
1→J
MAL(J)→MALO(J)
MTRF(J)→MTRFO(J)
J = 3
N
J+1→J
Y
NR1→NR10
NR2→NR20
RNG(0)→RN
RNG(NR2)→RN
31
NEXT PAGE B

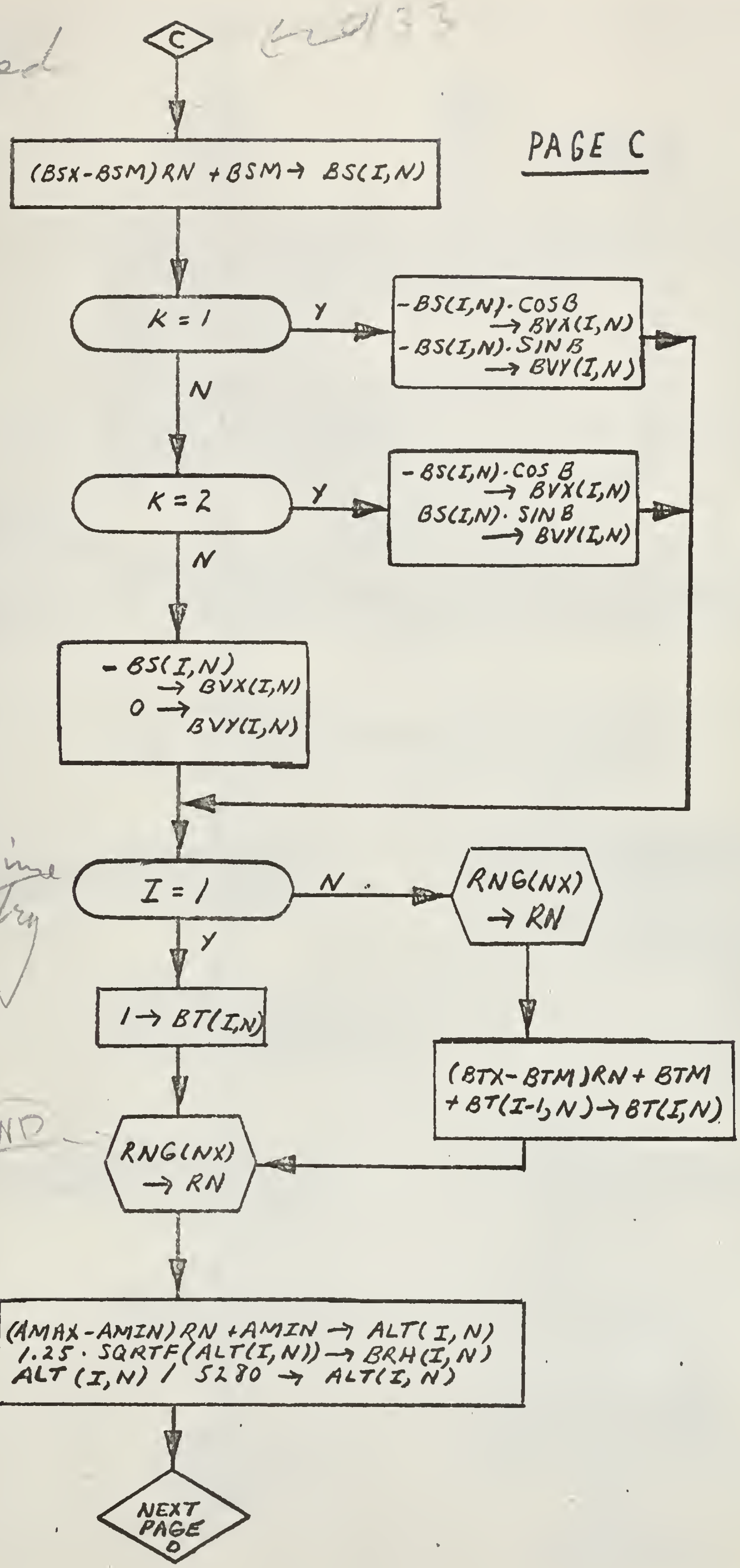
Boman entry

PAGE B



RND

Bomer Speed at Entry



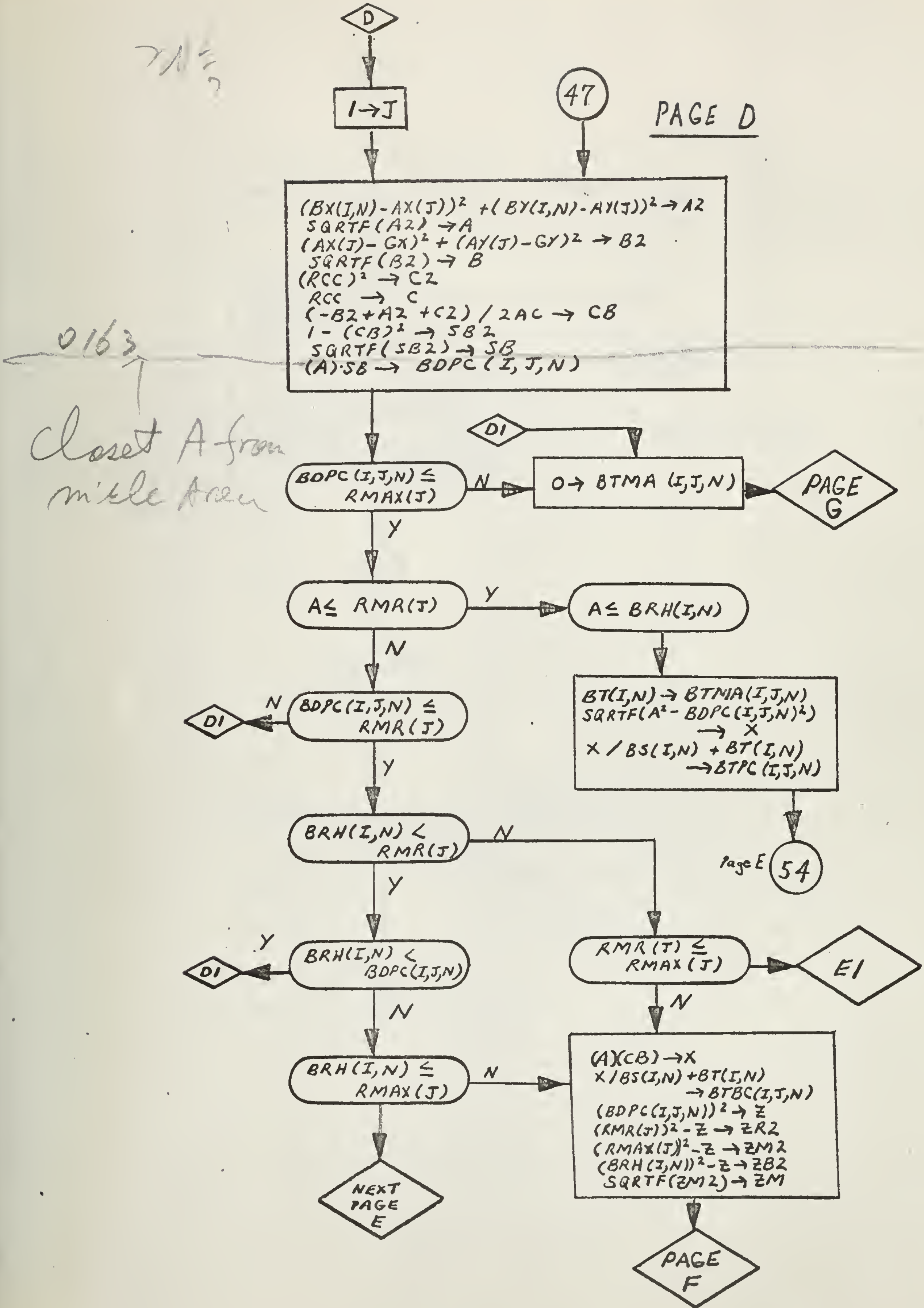
014C

Bomer time at entry

RND

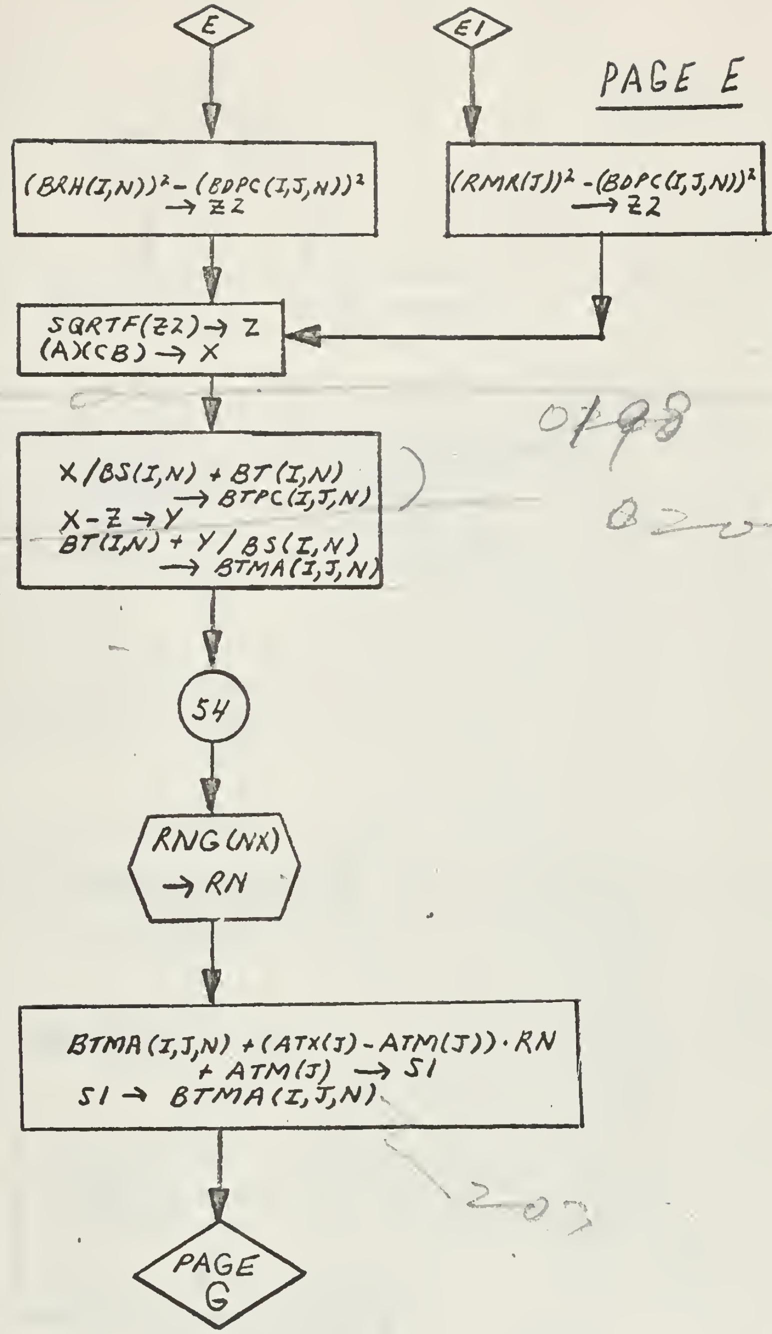
014P

Bomer altitude & Bomer Radar horizon



D
I→J
47
PAGE D
(BX(I,N)-AX(J))² +(BY(I,N)-AY(J))² → A2
SQRTF(A2) → A
(AX(J)-GX)² + (AY(J)-GY)² → B2
SQRTF(B2) → B
(RCC)² → C2
RCC → C
(-B2+A2+C2)/2AC → CB
1-(CB)² → SB2
SQRTF(SB2) → SB
(A)·SB → BDPC(I,J,N)
0163
Closest A from middle Area
D1
BDPC(I,J,N) ≤ RMAX(J)
N
0 → BTMA(I,J,N)
PAGE G
Y
A ≤ RMR(J)
A ≤ BRH(I,N)
BT(I,N) → BTMA(I,J,N)
SQRTF(A² - BDPC(I,J,N)²) → X
X/BS(I,N) + BT(I,N) → BTPC(I,J,N)
Page E
54
BDPC(I,J,N) ≤ RMR(J)
BRH(I,N) < RMR(J)
BRH(I,N) < BDPC(I,J,N)
RMR(J) ≤ RMAX(J)
E1
BRH(I,N) ≤ RMAX(J)
(A)(CB) → X
X/BS(I,N) + BT(I,N) → BTBC(I,J,N)
(BDPC(I,J,N))² → Z
(RMR(J))² - Z → ZR2
(RMAX(J))² - Z → ZM2
(BRH(I,N))² - Z → ZB2
SQRTF(ZM2) → ZM
NEXT PAGE E
PAGE F

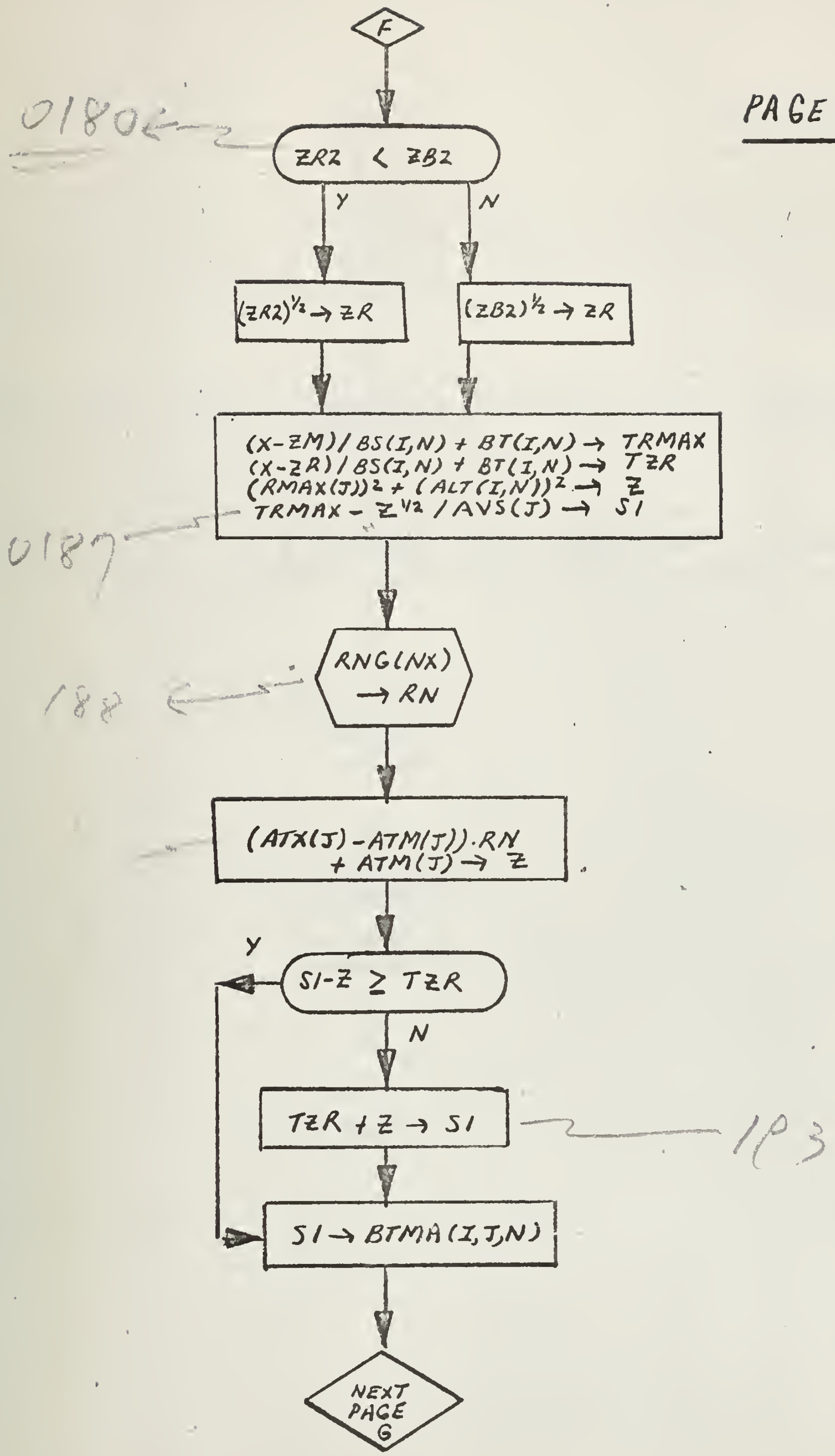
PAGE E



Time at pt of closest appr

The inst area entry

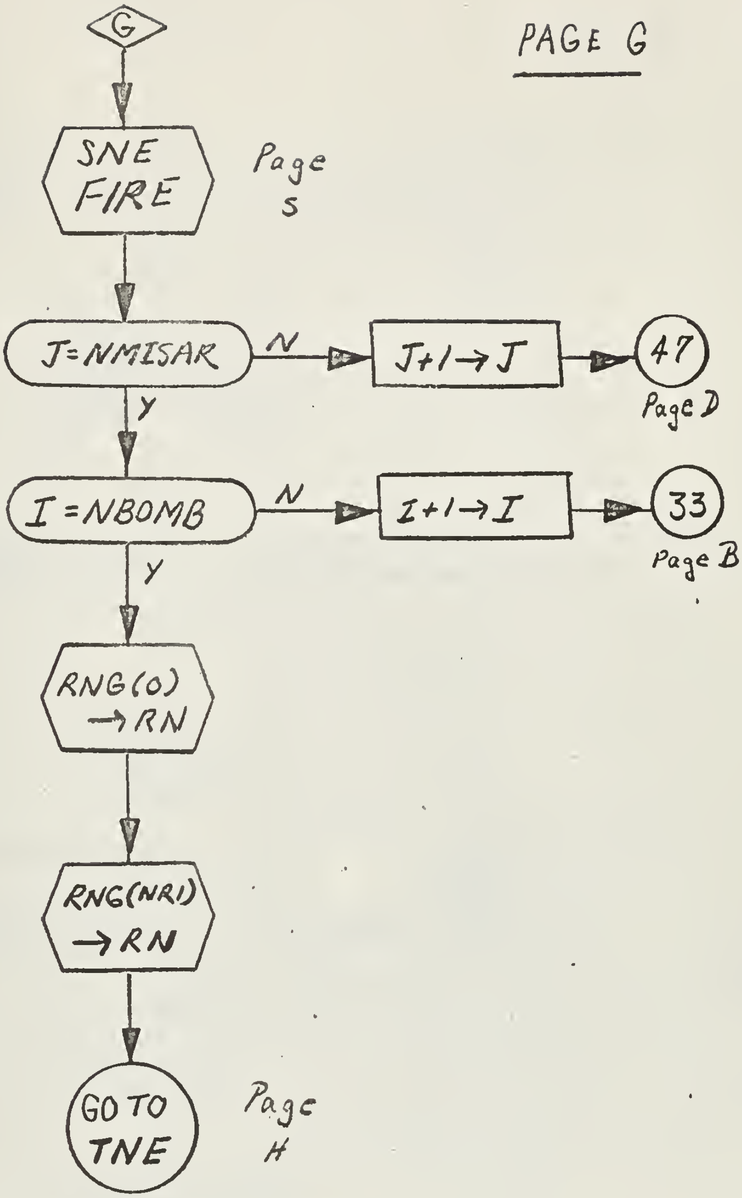
PAGE F

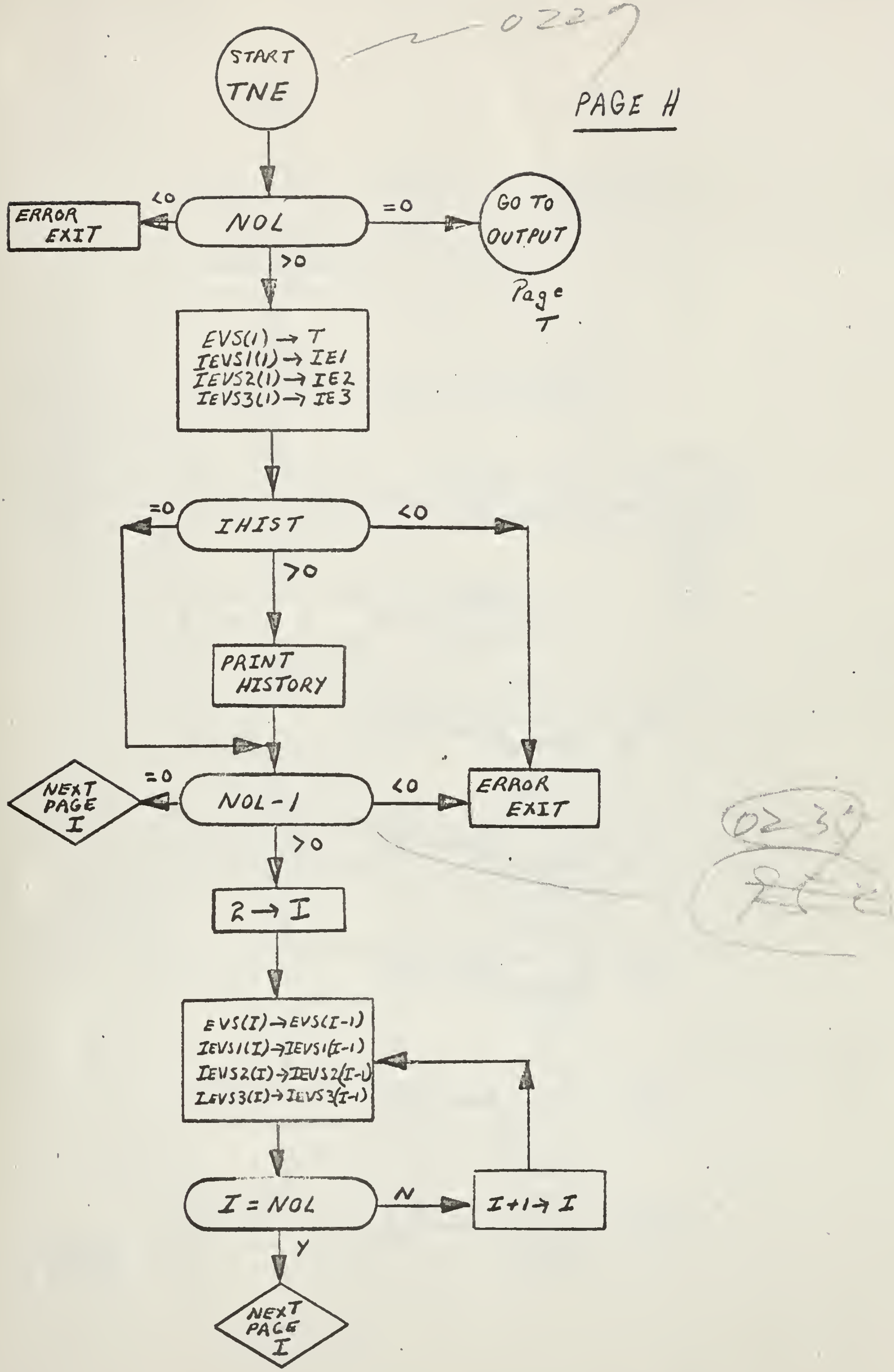


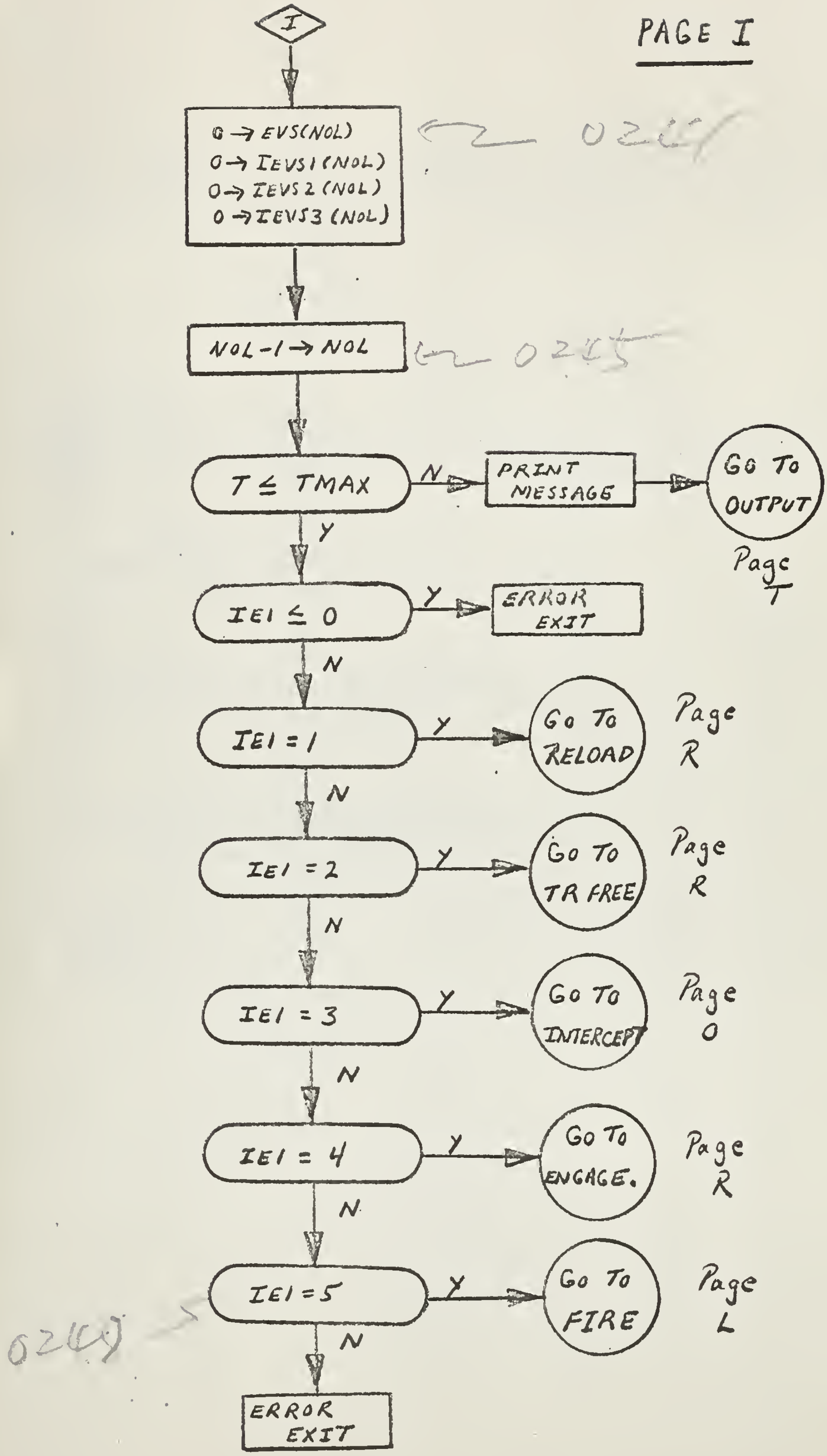
F
0180
ZR2 < ZB2
Y
N
(ZR2)^1/2 → ZR
(ZB2)^1/2 → ZR
(X-ZM)/BS(I,N) + BT(I,N) → TRMAX
(X-ZR)/BS(I,N) + BT(I,N) → TZR
(RMAX(J))^2 + (ALT(I,N))^2 → Z
TRMAX - Z^1/2 /AVS(J) → S1
0187
RNG(NX) → RN
188
(ATX(J)-ATM(J))·RN + ATM(J) → Z
Y
S1-Z ≥ TZR
N
TZR + Z → S1
183
S1 → BTMA(I,J,N)
NEXT PAGE G

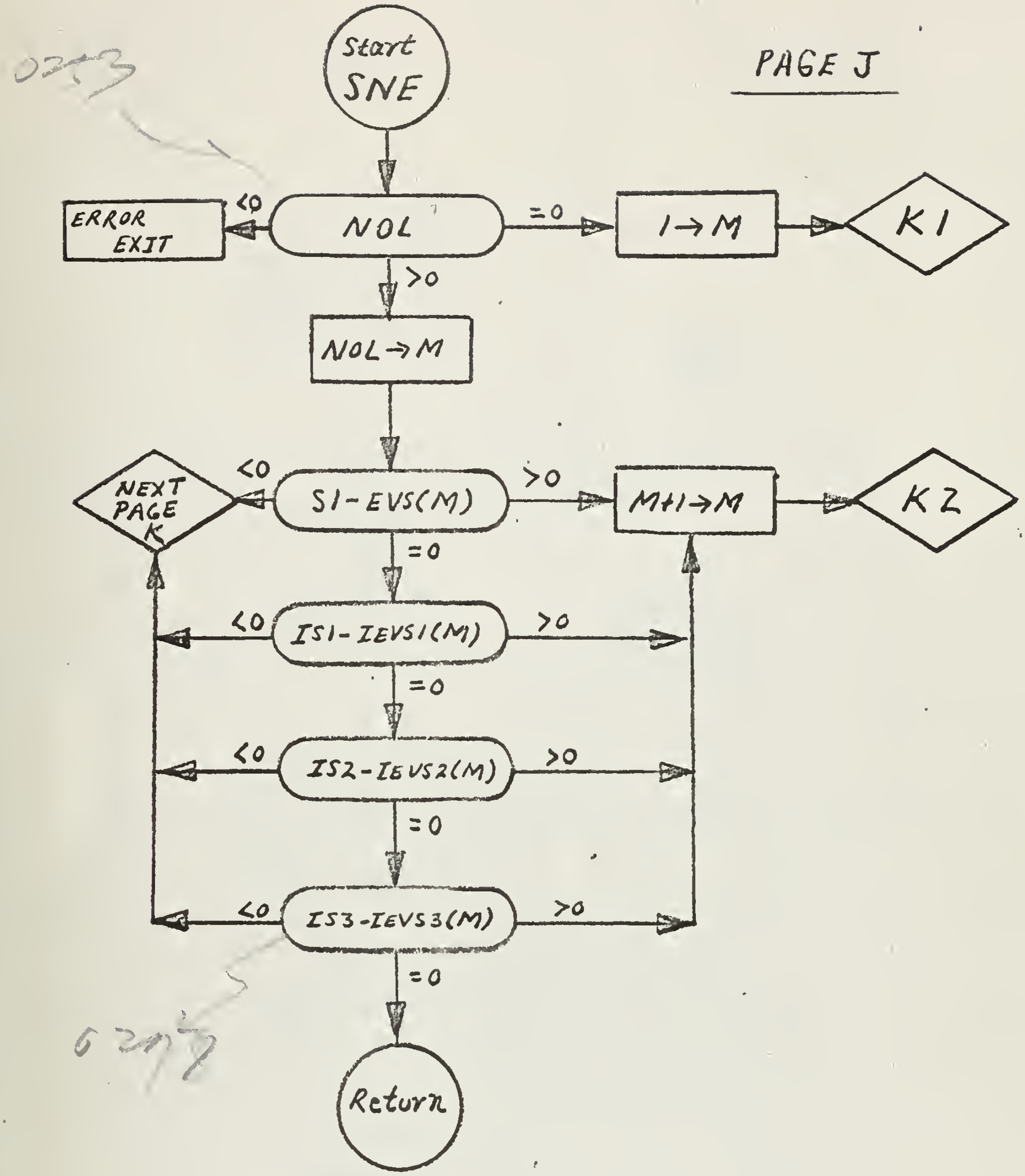
PAGE G

0217

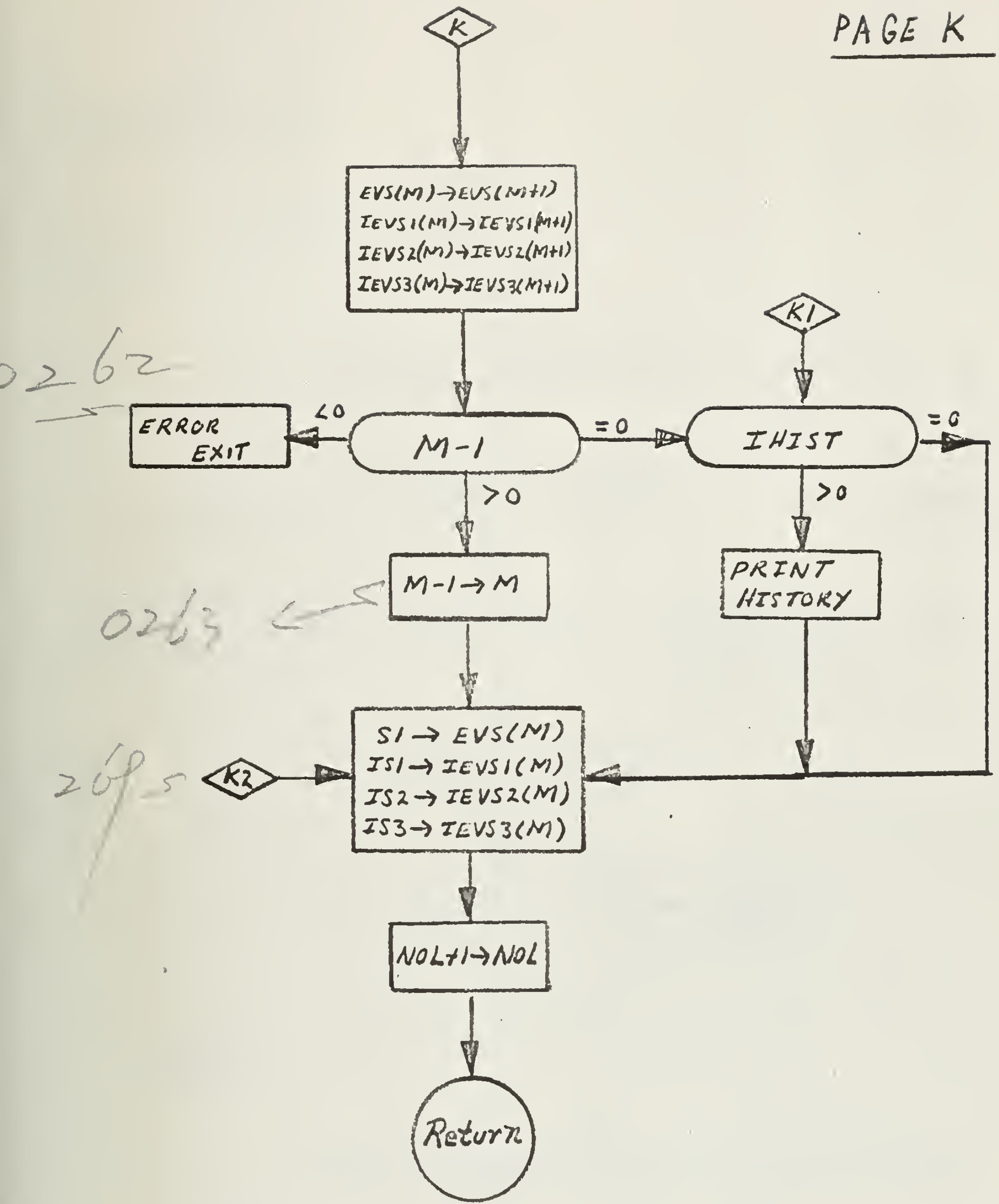

G
SNE FIRE
Page S
J=NMISAR
N
J+1→J
47
Page D
Y
I=NBOMB
N
I+1→I
33
Page B
Y
RNG(0) →RN
RNG(NRI) →RN
GO TO TNE
Page H


PAGE H
START TNE
NOL
<0
ERROR EXIT
=0
GO TO OUTPUT
Page T
>0
EVS(1) → T
IEVS1(1) → IE1
IEVS2(1) → IE2
IEVS3(1) → IE3
IHIST
=0
<0
>0
PRINT HISTORY
NOL-1
=0
NEXT PAGE I
<0
ERROR EXIT
>0
2 → I
EVS(I) → EVS(I-1)
IEVS1(I) → IEVS1(I-1)
IEVS2(I) → IEVS2(I-1)
IEVS3(I) → IEVS3(I-1)
I = NOL
N
I+1 → I
Y
NEXT PAGE I

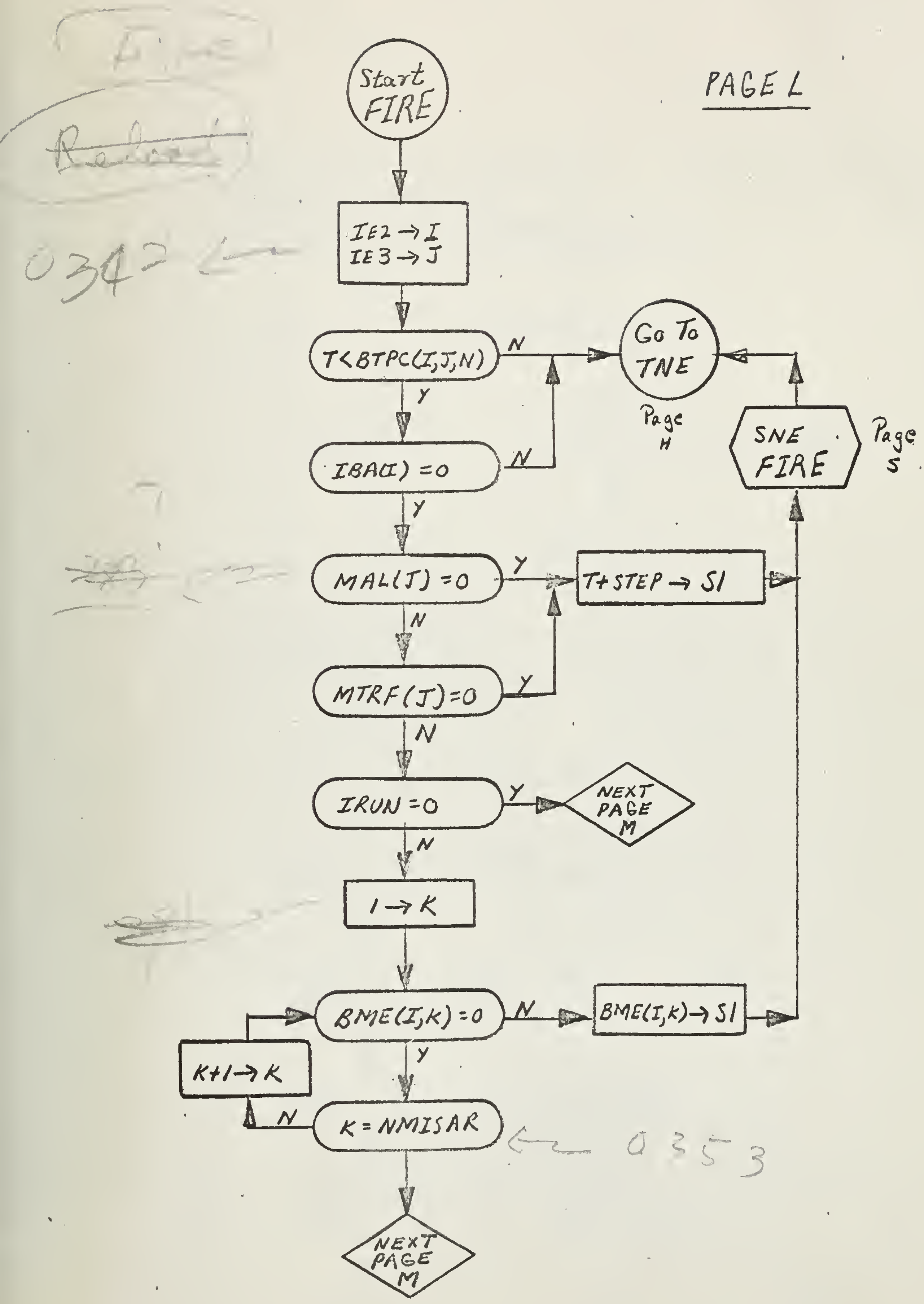

I
PAGE I
0 → EVS(NOL)
0 → IEVS1(NOL)
0 → IEVS2(NOL)
0 → IEVS3(NOL)
0Z(?)
NOL-1 → NOL
0Z(5)
T ≤ TMAX
N
PRINT MESSAGE
GO TO OUTPUT
Page T
Y
IE1 ≤ 0
ERROR EXIT
IE1 = 1
GO TO RELOAD
Page R
IE1 = 2
GO TO TR FREE
Page R
IE1 = 3
GO TO INTERCEPT
Page O
IE1 = 4
GO TO ENGAGE.
Page R
IE1 = 5
GO TO FIRE
Page L
0Z(?)
ERROR EXIT


PAGE J
Start
SNE
ERROR
EXIT
<0
NOL
=0
1→M
K1
>0
NOL→M
NEXT
PAGE
K
<0
S1-EVS(M)
>0
M+1→M
K2
=0
<0
IS1-IEVS1(M)
>0
=0
<0
IS2-IEVS2(M)
>0
=0
<0
IS3-IEVS3(M)
>0
=0
Return
0253
0257

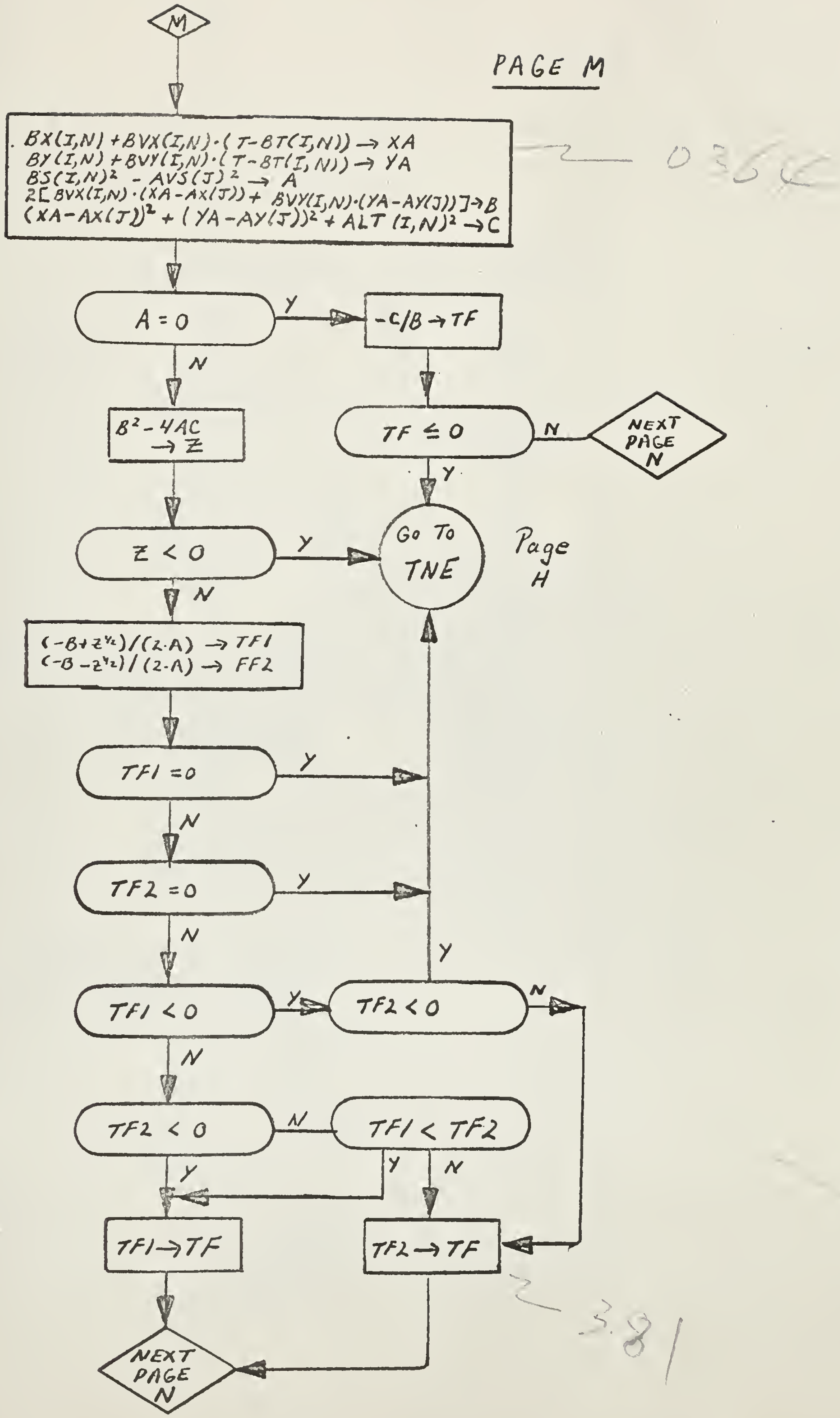
PAGE K



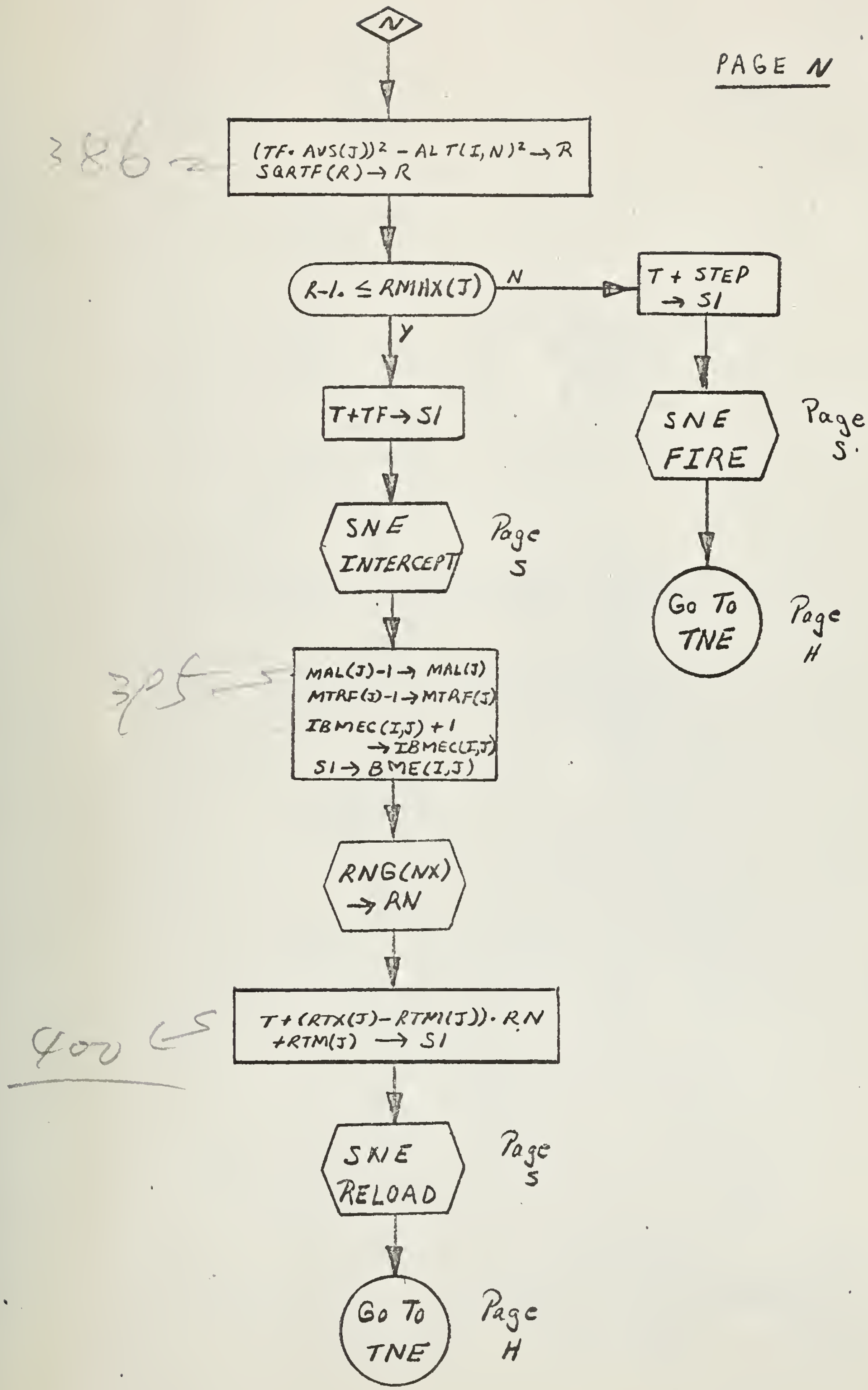
K
EVS(M)→EVS(M+1)
IEVS1(M)→IEVS1(M+1)
IEVS2(M)→IEVS2(M+1)
IEVS3(M)→IEVS3(M+1)
K1
0262
ERROR EXIT
<0
M-1
=0
IHIST
=0
>0
>0
M-1→M
PRINT HISTORY
0263
S1→EVS(M)
IS1→IEVS1(M)
IS2→IEVS2(M)
IS3→IEVS3(M)
K2
NOL+1→NOL
Return



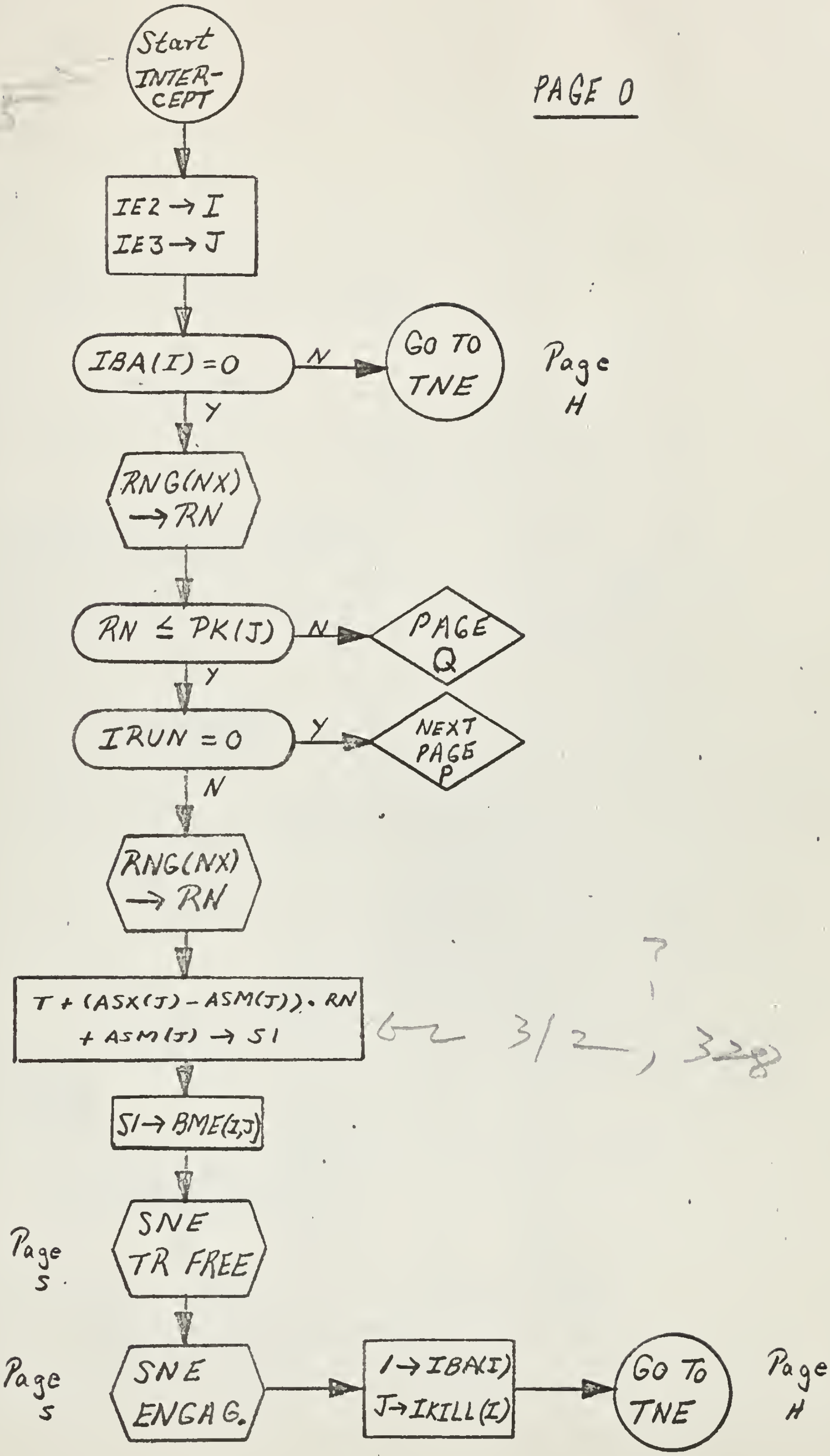
PAGE L
Start FIRE
IE2 → I
IE3 → J
T<BTPC(I,J,N)
N
Y
Go To TNE
Page H
IBA(I) = 0
N
Y
SNE FIRE
Page S
MAL(J) = 0
Y
N
T+STEP → SI
MTRF(J)=0
Y
N
IRUN = 0
Y
NEXT PAGE M
N
1 → K
BME(I,K) = 0
N
BME(I,K) → SI
Y
K+1 → K
K = NMISAR
N
0353
NEXT PAGE M
0342


M
PAGE M
BX(I,N) +BVX(I,N)·(T-BT(I,N)) → XA
BY(I,N) +BVY(I,N)·(T-BT(I,N)) → YA
BS(I,N)² - AVS(J)² → A
2[BVX(I,N)·(XA-AX(J)) + BVY(I,N)·(YA-AY(J))] → B
(XA-AX(J))² + (YA-AY(J))² + ALT (I,N)² → C
0366
A = 0
Y
N
-C/B → TF
B² - 4AC → Z
TF ≤ 0
NEXT PAGE N
Z < 0
Go To TNE
Page H
(-B+Z^½)/(2·A) → TF1
(-B-Z^½)/(2·A) → FF2
TF1 = 0
TF2 = 0
TF1 < 0
TF2 < 0
TF2 < 0
TF1 < TF2
TF1 → TF
TF2 → TF
381
NEXT PAGE N

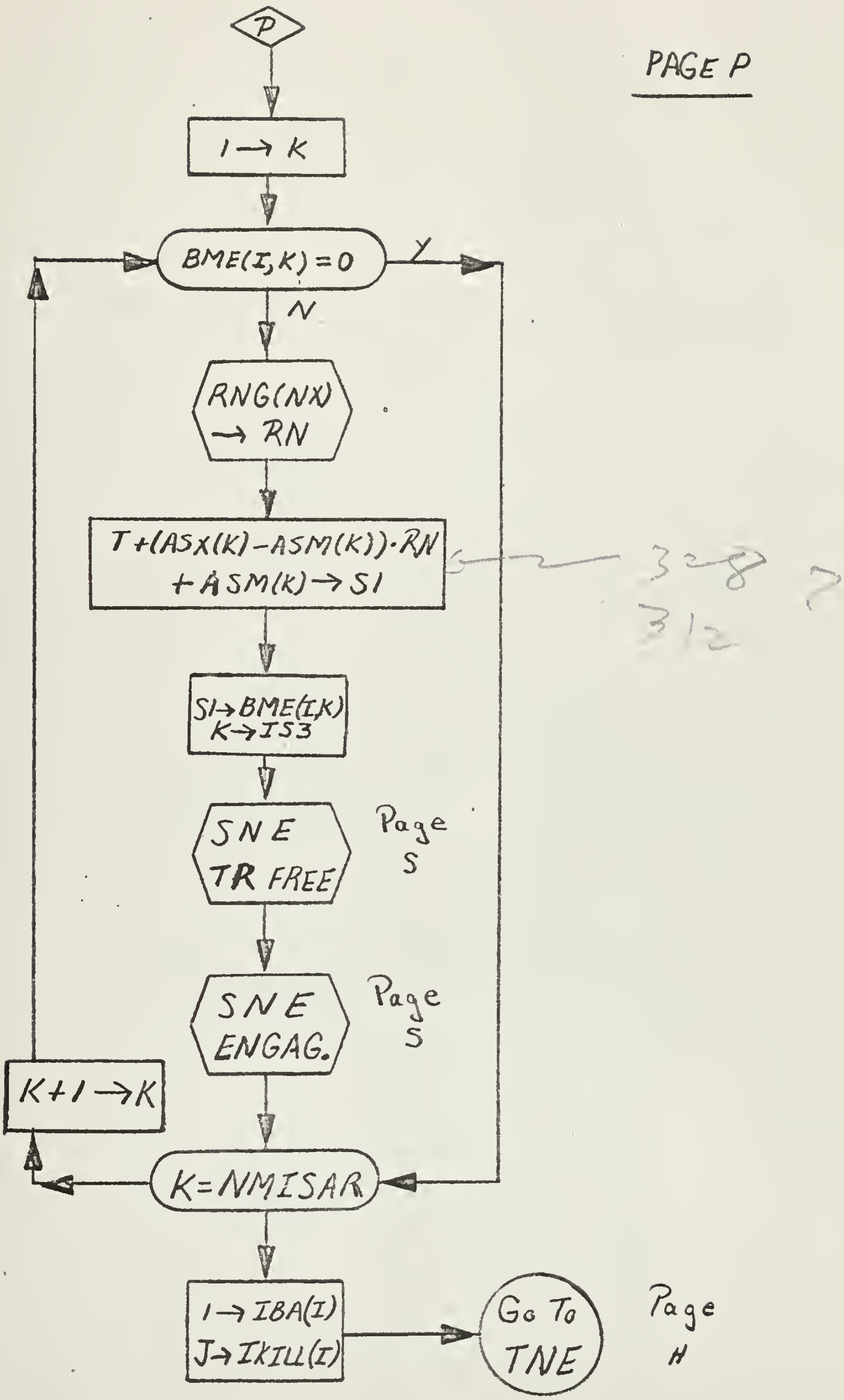
PAGE N



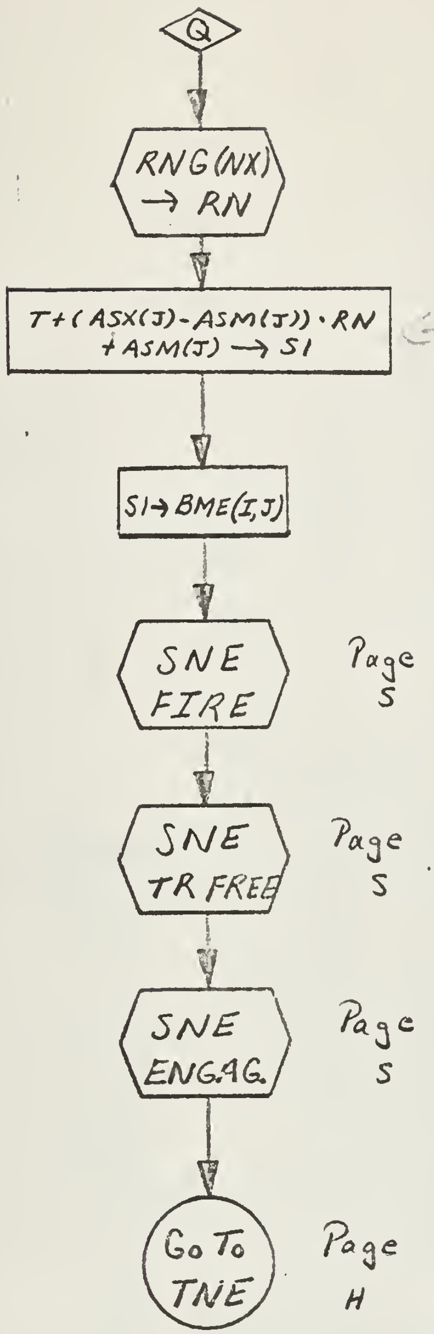
N
386
(TF· AVS(J))² − ALT(I,N)² → R
SQRTF(R) → R
R-1. ≤ RMAX(J)
N
Y
T + STEP → S1
T+TF → S1
SNE FIRE
Page S.
SNE INTERCEPT
Page S
Go To TNE
Page H
395
MAL(J)-1 → MAL(J)
MTRF(J)-1 → MTRF(J)
IBMEC(I,J) + 1 → IBMEC(I,J)
S1 → BME(I,J)
RNG(NX) → RN
400
T+(RTX(J)−RTM(J))·RN +RTM(J) → S1
SNE RELOAD
Page S
Go To TNE
Page H


Start INTERCEPT
IE2 → I
IE3 → J
IBA(I) = 0
N
Y
GO TO TNE
Page H
RNG(NX) → RN
RN ≤ PK(J)
N
Y
PAGE Q
IRUN = 0
Y
N
NEXT PAGE P
RNG(NX) → RN
T + (ASX(J) − ASM(J)) · RN + ASM(J) → S1
S1 → BME(I,J)
Page S
SNE TR FREE
Page S
SNE ENGAG.
1 → IBA(I)
J → IKILL(I)
GO TO TNE
Page H

PAGE P

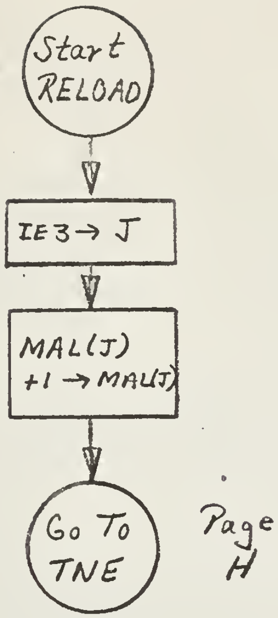

P
1 → K
BME(I,K) = 0
Y
N
RNG(NX) → RN
T+(ASX(K)−ASM(K))·RN + ASM(K) → SI
328 ?
312
SI → BME(I,K)
K → IS3
SNE TR FREE
Page S
SNE ENGAG.
Page S
K+1 → K
K = NMISAR
1 → IBA(I)
J → IKILL(I)
Go To TNE
Page H

PAGE Q

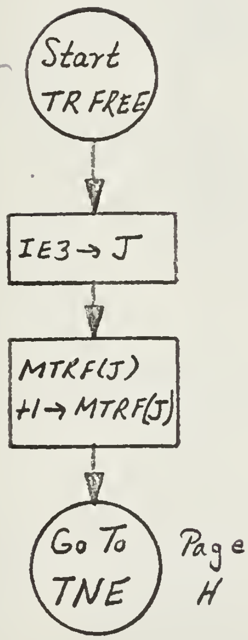

Q
RNG(NX) → RN
T+(ASX(J)-ASM(J))·RN + ASM(J) → S1
S1→BME(I,J)
SNE FIRE
Page S
SNE TR FREE
Page S
SNE ENGAG.
Page S
Go To TNE
Page H

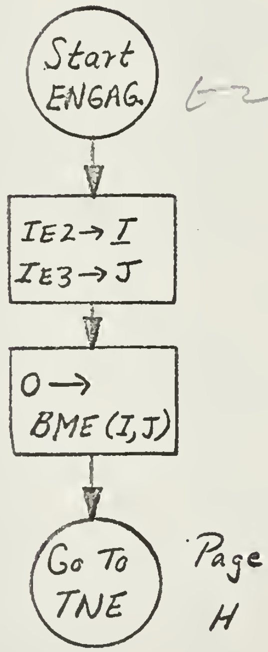
217

PAGE R

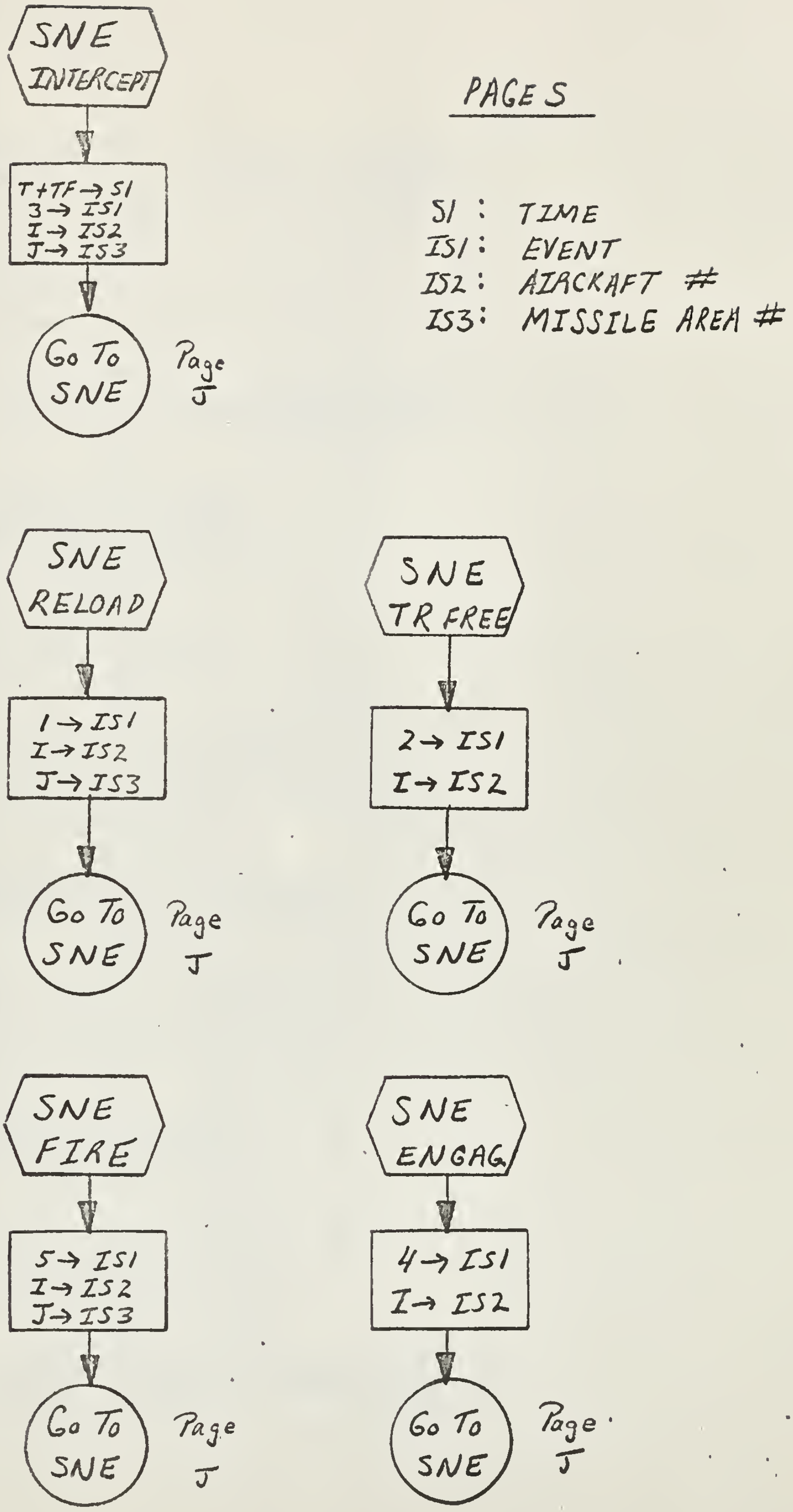


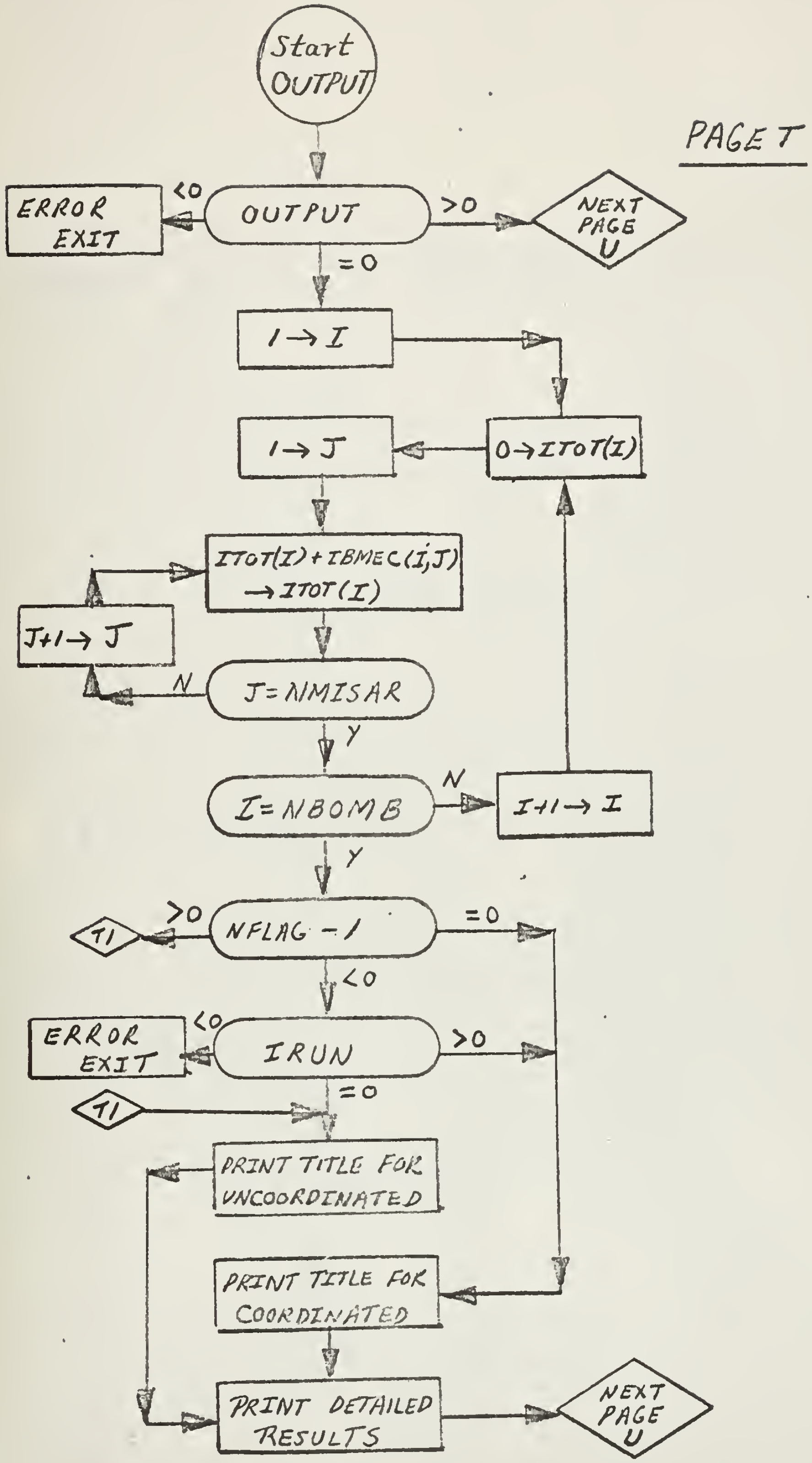
282



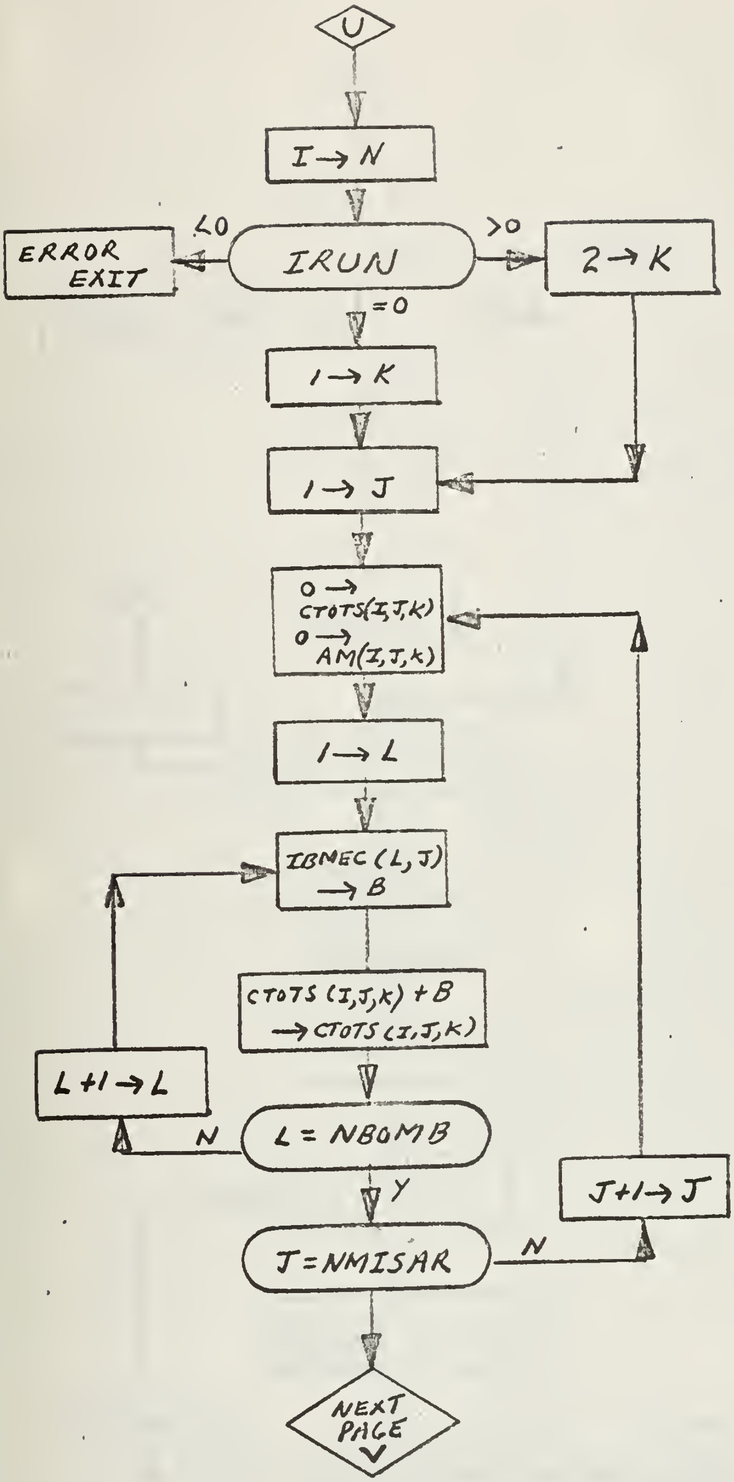


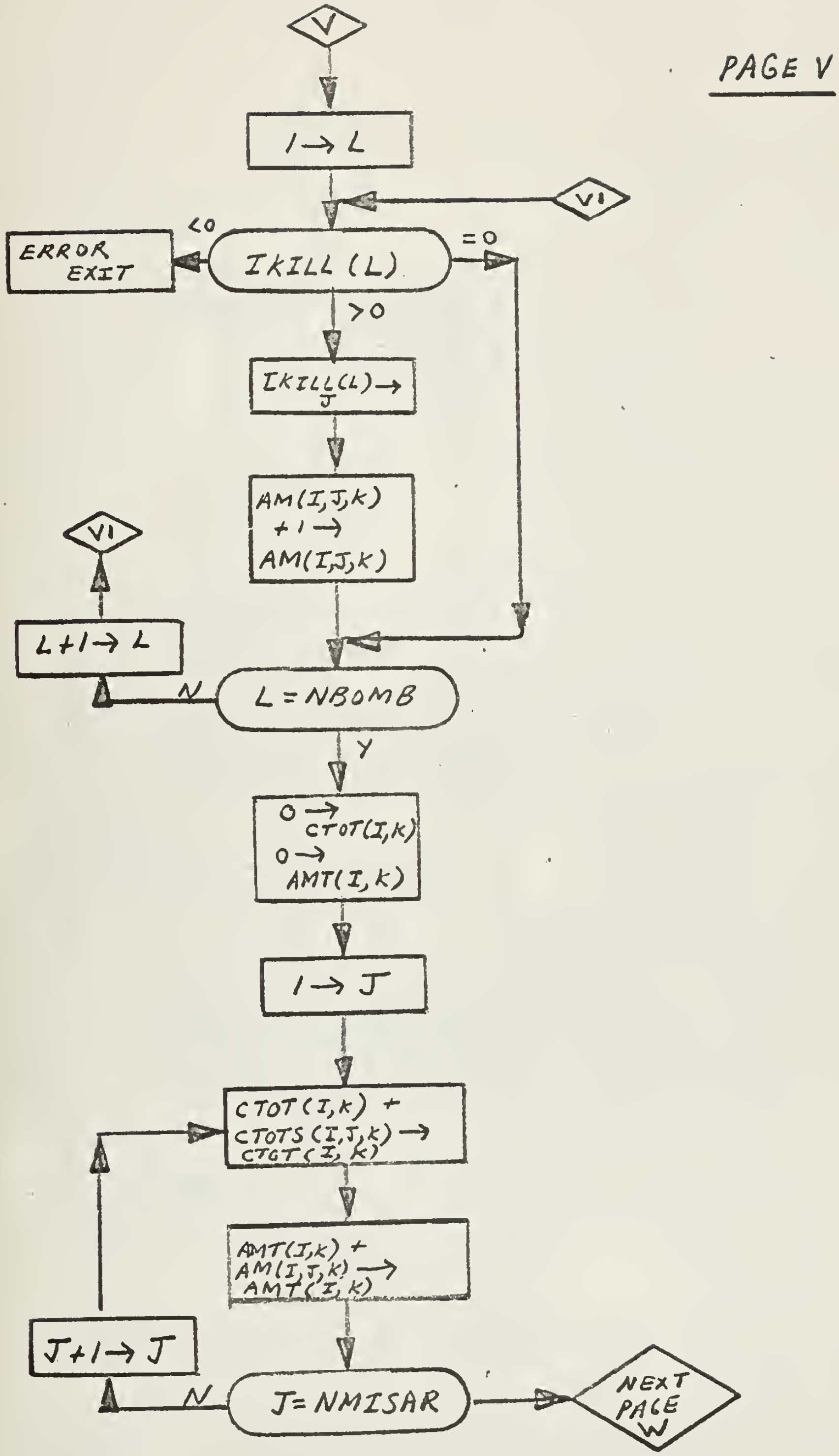
338

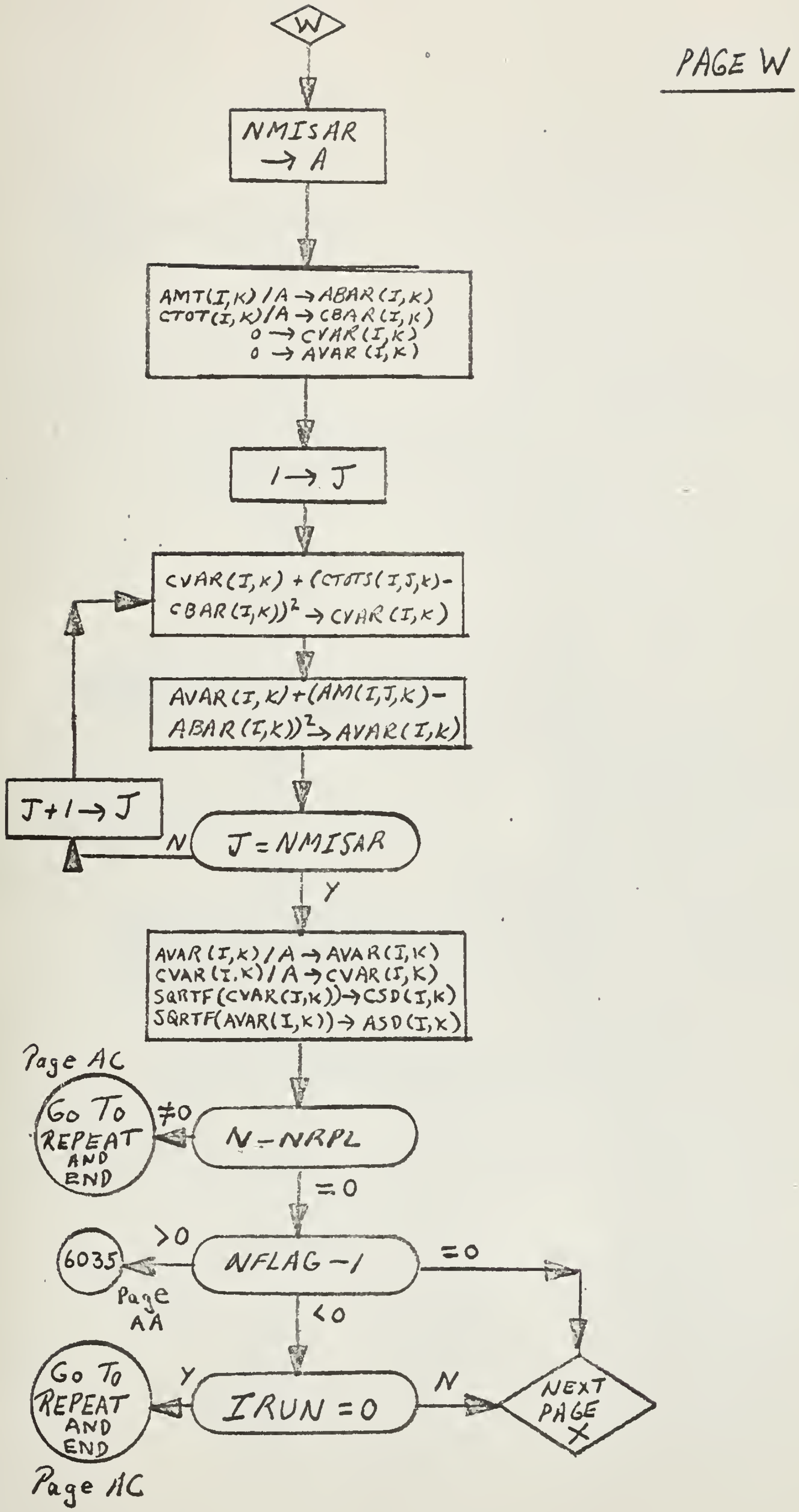

PAGE S
S1 : TIME
IS1 : EVENT
IS2 : AIRCRAFT #
IS3 : MISSILE AREA #
SNE INTERCEPT
T+TF → S1
3 → IS1
I → IS2
J → IS3
Go To SNE
Page J
SNE RELOAD
1 → IS1
I → IS2
J → IS3
Go To SNE
Page J
SNE TR FREE
2 → IS1
I → IS2
Go To SNE
Page J
SNE FIRE
5 → IS1
I → IS2
J → IS3
Go To SNE
Page J
SNE ENGAG
4 → IS1
I → IS2
Go To SNE
Page J

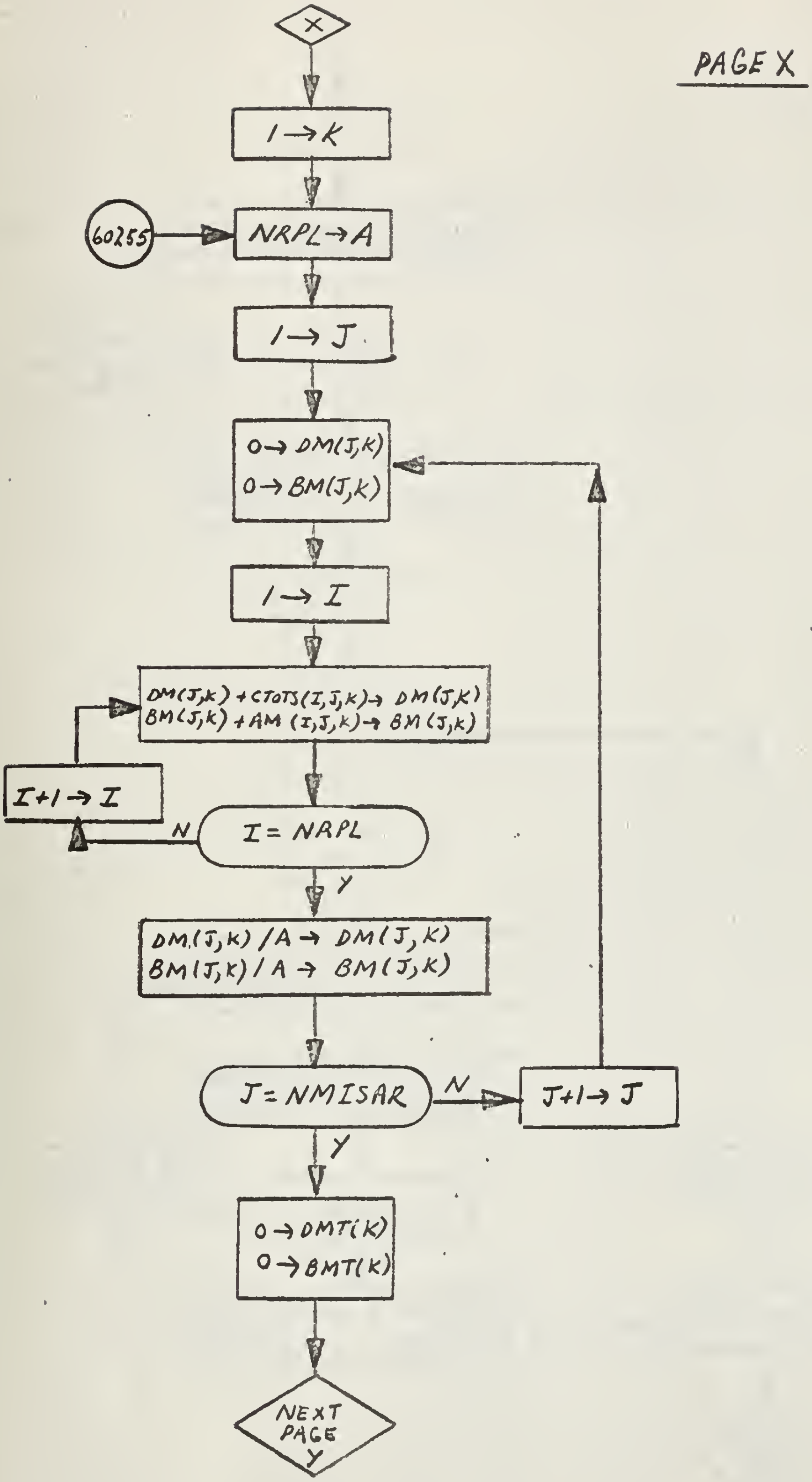

PAGE T
Start OUTPUT
OUTPUT
<0
>0
=0
ERROR EXIT
NEXT PAGE U
1 → I
1 → J
0 → ITOT(I)
ITOT(I) + IBMEC(I,J) → ITOT(I)
J+1 → J
J = NMISAR
N
Y
I = NBOMB
I+1 → I
NFLAG − 1
T1
IRUN
PRINT TITLE FOR UNCOORDINATED
PRINT TITLE FOR COORDINATED
PRINT DETAILED RESULTS

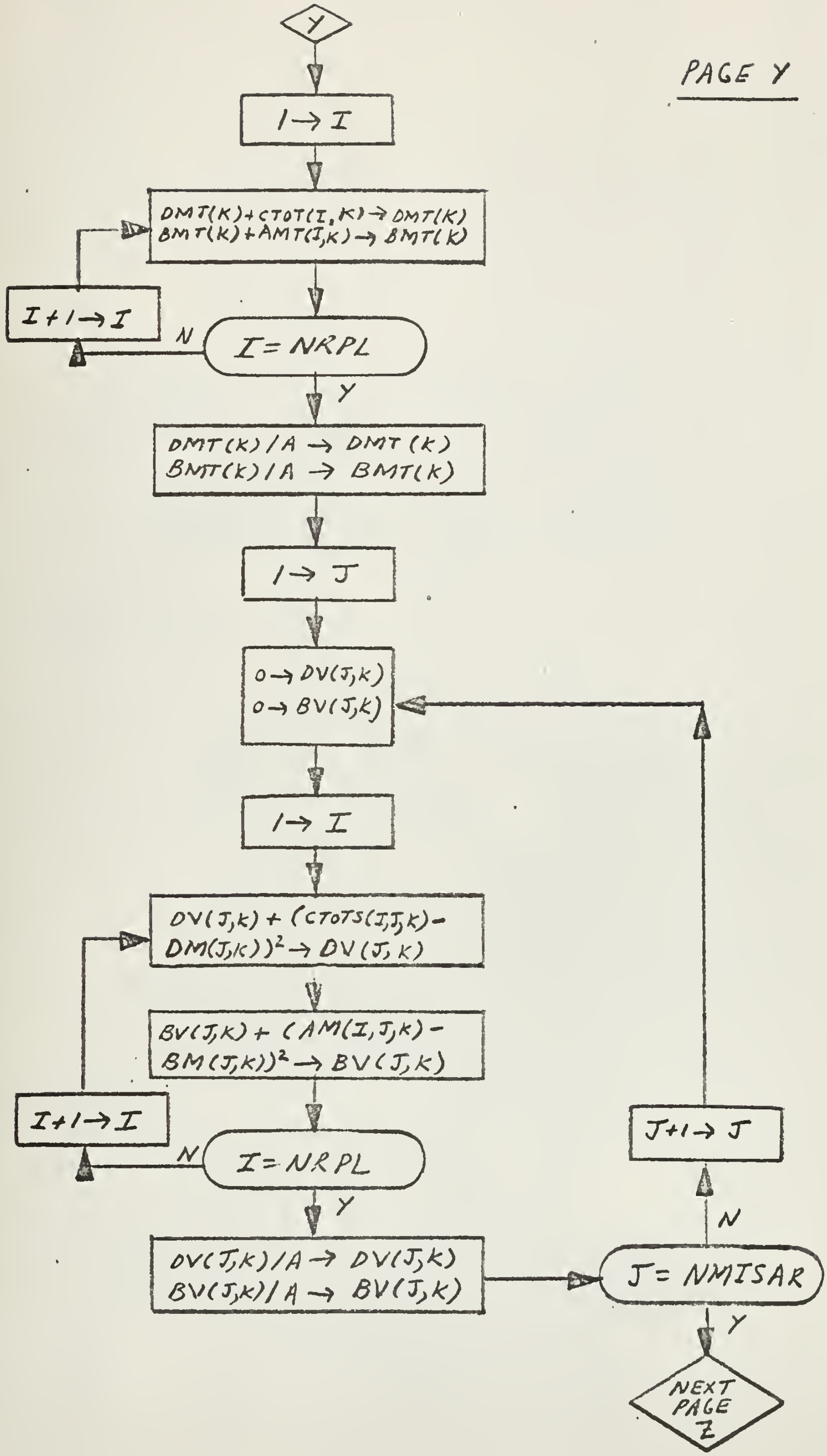
PAGE U

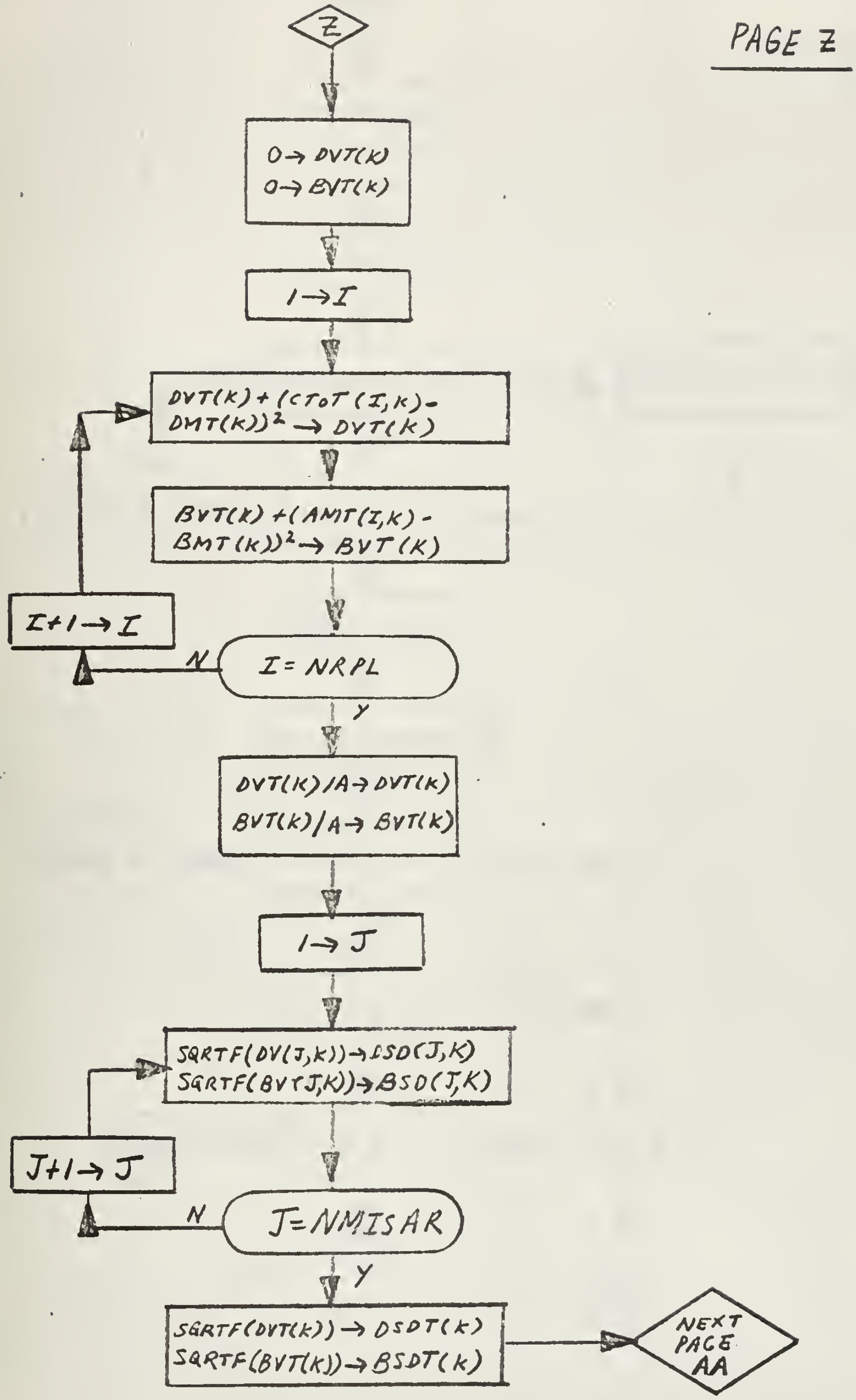

U
I → N
IRUN
<0
ERROR EXIT
>0
2 → K
=0
1 → K
1 → J
0 → CTOTS(I,J,K)
0 → AM(I,J,K)
1 → L
IBNEC (L,J) → B
CTOTS (I,J,K) + B → CTOTS (I,J,K)
L+1 → L
N
L = NBOMB
Y
J = NMISAR
N
J+1 → J
NEXT PAGE
V


PAGE V
V
1 → L
VI
<0
=0
ERROR EXIT
IKILL (L)
>0
IKILL(L) → J
AM(I,J,K) +1 → AM(I,J,K)
VI
L+1 → L
N
L = NBOMB
Y
0 → CTOT(I,K)
0 → AMT(I,K)
1 → J
CTOT(I,K) + CTOTS(I,J,K) → CTOT(I,K)
AMT(I,K) + AM(I,J,K) → AMT(I,K)
J+1 → J
N
J = NMISAR
NEXT PAGE W


PAGE W
W
NMISAR → A
AMT(I,K)/A → ABAR(I,K)
CTOT(I,K)/A → CBAR(I,K)
0 → CVAR(I,K)
0 → AVAR(I,K)
1 → J
CVAR(I,K) + (CTOTS(I,J,K)-CBAR(I,K))² → CVAR(I,K)
AVAR(I,K)+(AM(I,J,K)-ABAR(I,K))² → AVAR(I,K)
J+1 → J
N
J = NMISAR
Y
AVAR(I,K)/A → AVAR(I,K)
CVAR(I,K)/A → CVAR(I,K)
SQRTF(CVAR(I,K)) → CSD(I,K)
SQRTF(AVAR(I,K)) → ASD(I,K)
Page AC
Go To REPEAT AND END
≠0
N-NRPL
=0
>0
6035
Page AA
NFLAG-1
=0
<0
Go To REPEAT AND END
Y
IRUN = 0
N
NEXT PAGE X
Page AC

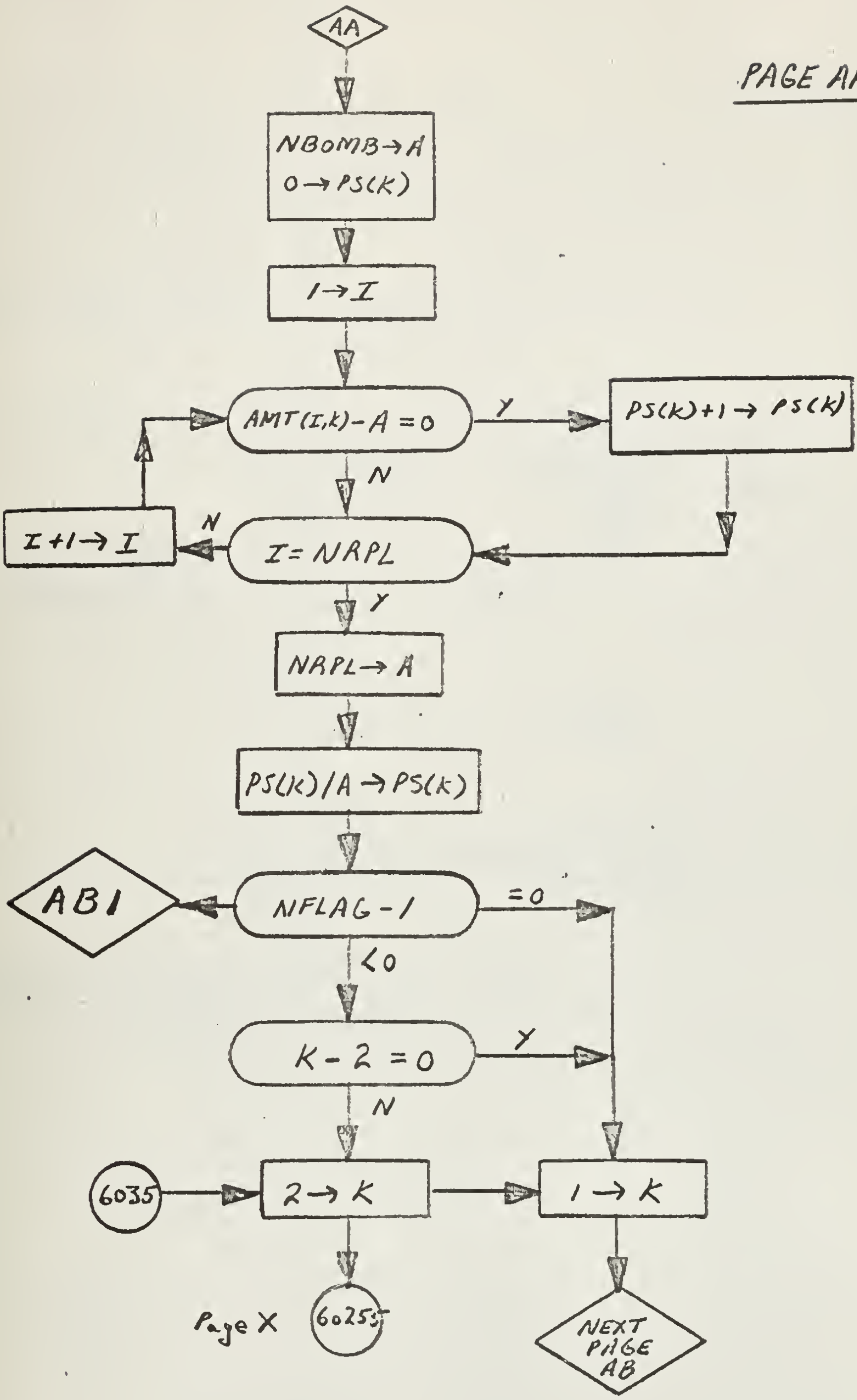

X
PAGE X
1→K
60255
NRPL→A
1→J
0→DM(J,K)
0→BM(J,K)
1→I
DM(J,K)+CTOTS(I,J,K)→DM(J,K)
BM(J,K)+AM(I,J,K)→BM(J,K)
I+1→I
N
I=NRPL
Y
DM(J,K)/A→DM(J,K)
BM(J,K)/A→BM(J,K)
J=NMISAR
N
J+1→J
Y
0→DMT(K)
0→BMT(K)
NEXT PAGE
Y


PAGE Y
Y
1→I
DMT(K)+CTOT(I,K)→DMT(K)
BMT(K)+AMT(I,K)→BMT(K)
I+1→I
N
I=NRPL
Y
DMT(K)/A → DMT(K)
BMT(K)/A → BMT(K)
1→J
0→DV(J,K)
0→BV(J,K)
1→I
DV(J,K)+(CTOTS(I,J,K)-DM(J,K))²→DV(J,K)
BV(J,K)+(AM(I,J,K)-BM(J,K))²→BV(J,K)
I+1→I
N
I=NRPL
Y
DV(J,K)/A→DV(J,K)
BV(J,K)/A→BV(J,K)
J=NMISAR
N
J+1→J
Y
NEXT PAGE Z



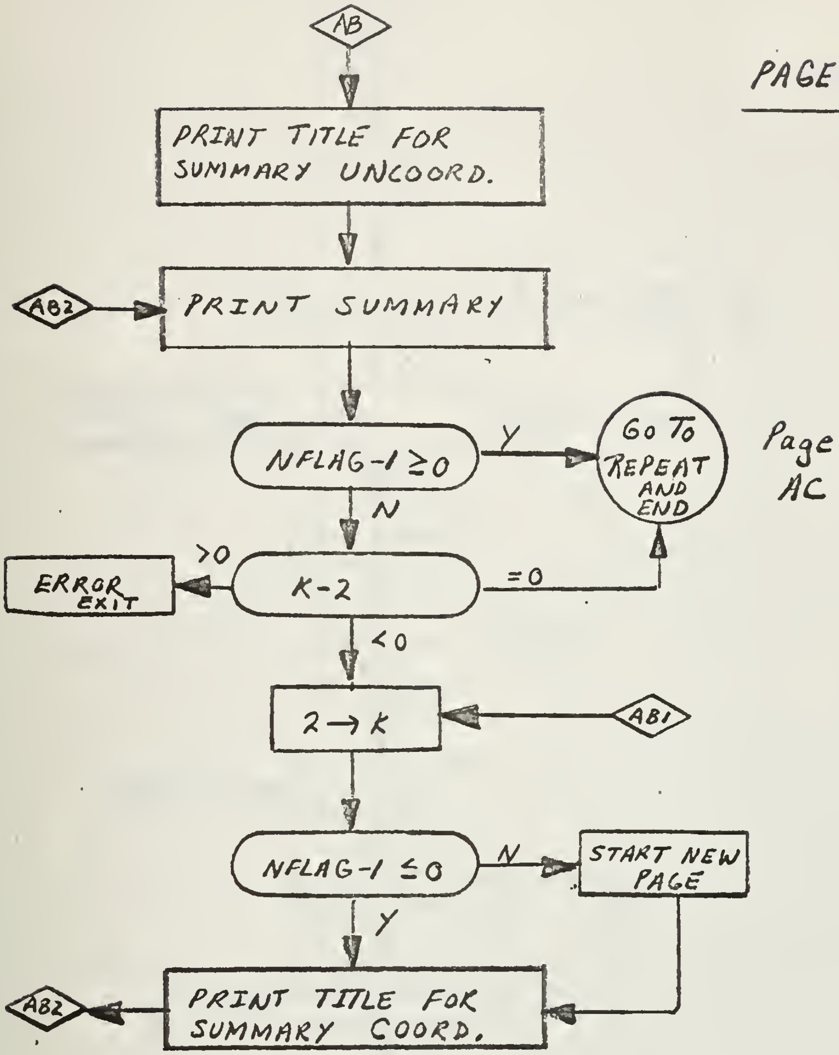
Z
PAGE Z
0→ DVT(K)
0→ BVT(K)
1→I
DVT(K) + (CTOT (I,K)-
DMT(K))² → DVT(K)
BVT(K) +(AMT(I,K) -
BMT (K))²→ BVT (K)
I+1→I
N
I = NRPL
Y
DVT(K)/A→ DVT(K)
BVT(K)/A→ BVT(K)
1→ J
SQRTF(DV(J,K))→DSD(J,K)
SQRTF(BV(J,K))→BSD(J,K)
J+1→ J
N
J=NMISAR
Y
SQRTF(DVT(K)) → DSDT(K)
SQRTF(BVT(K))→ BSDT(K)
NEXT
PAGE
AA

PAGE AA

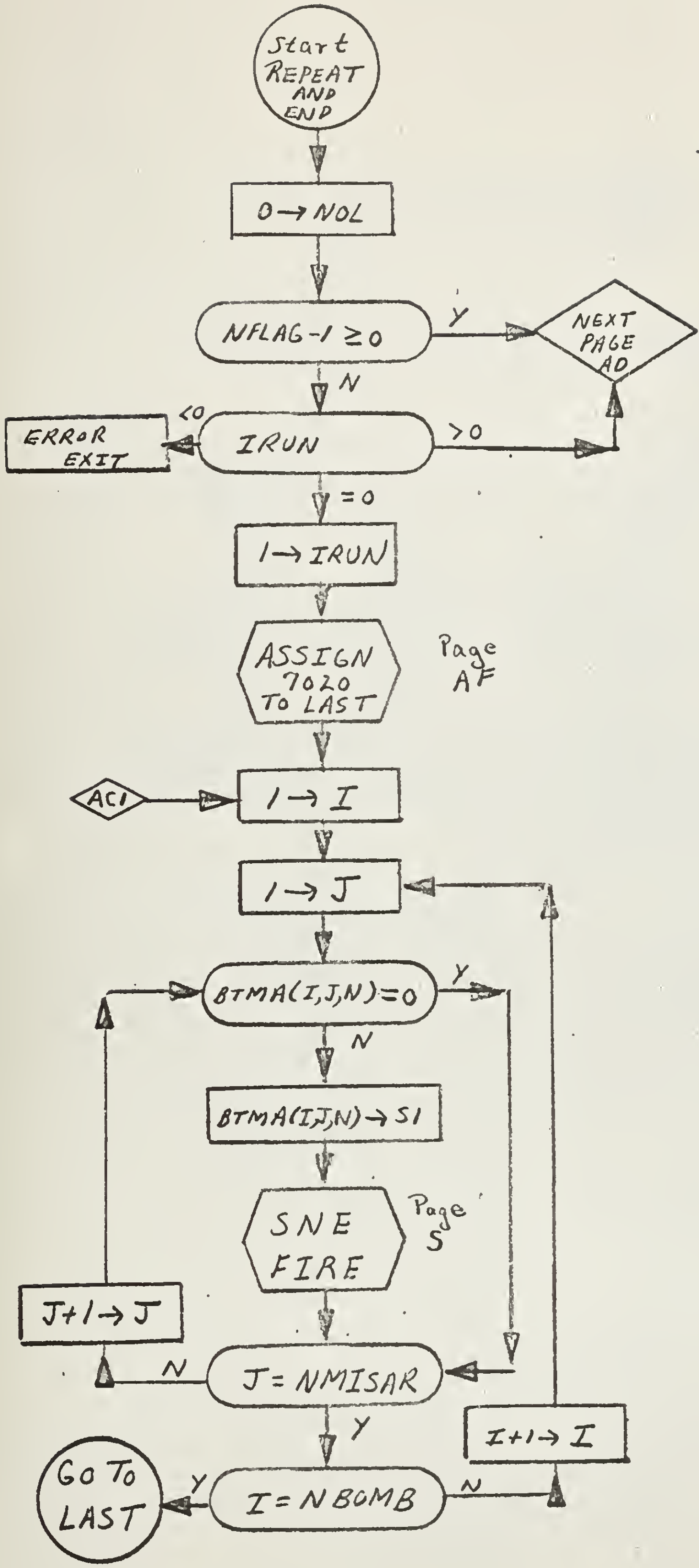


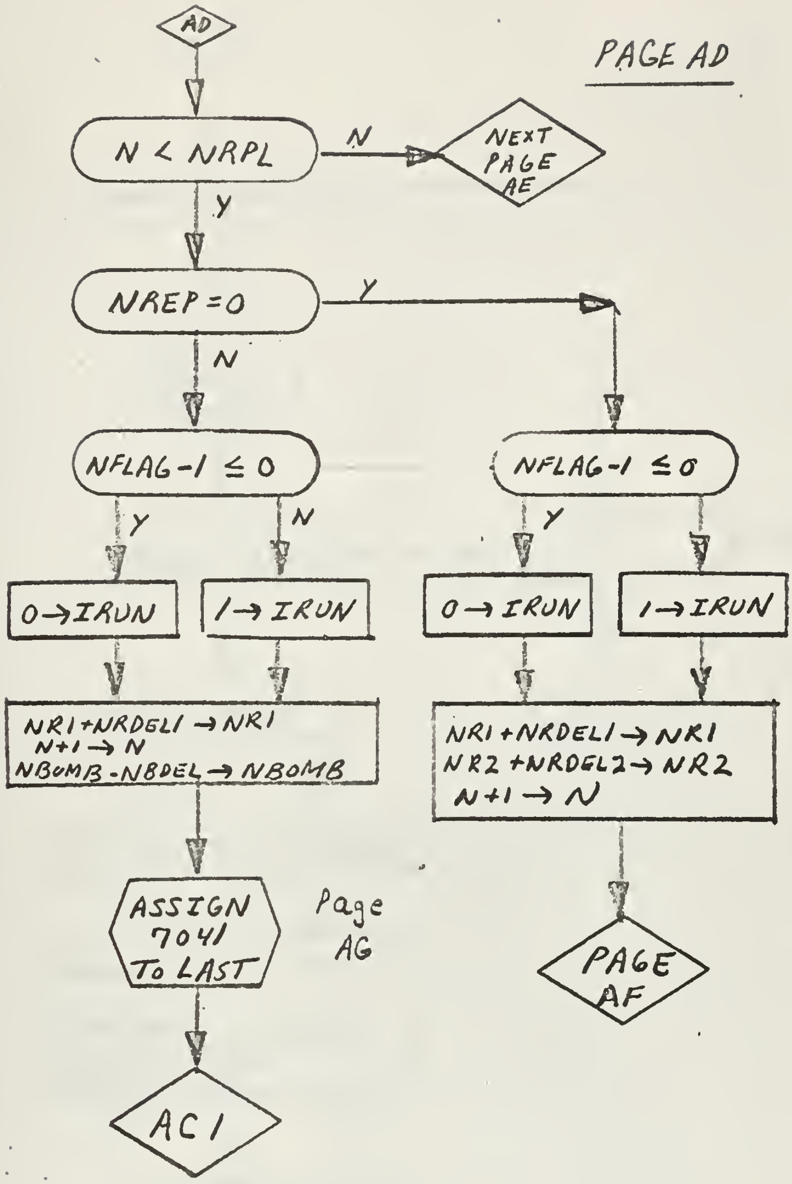
AA
NBOMB → A
0 → PS(K)
1 → I
AMT(I,K) - A = 0
Y
PS(K)+1 → PS(K)
N
I = NRPL
N
I+1 → I
Y
NRPL → A
PS(K)/A → PS(K)
ABI
NFLAG - 1
=0
<0
K - 2 = 0
Y
N
6035
2 → K
1 → K
Page X
6025
NEXT PAGE AB

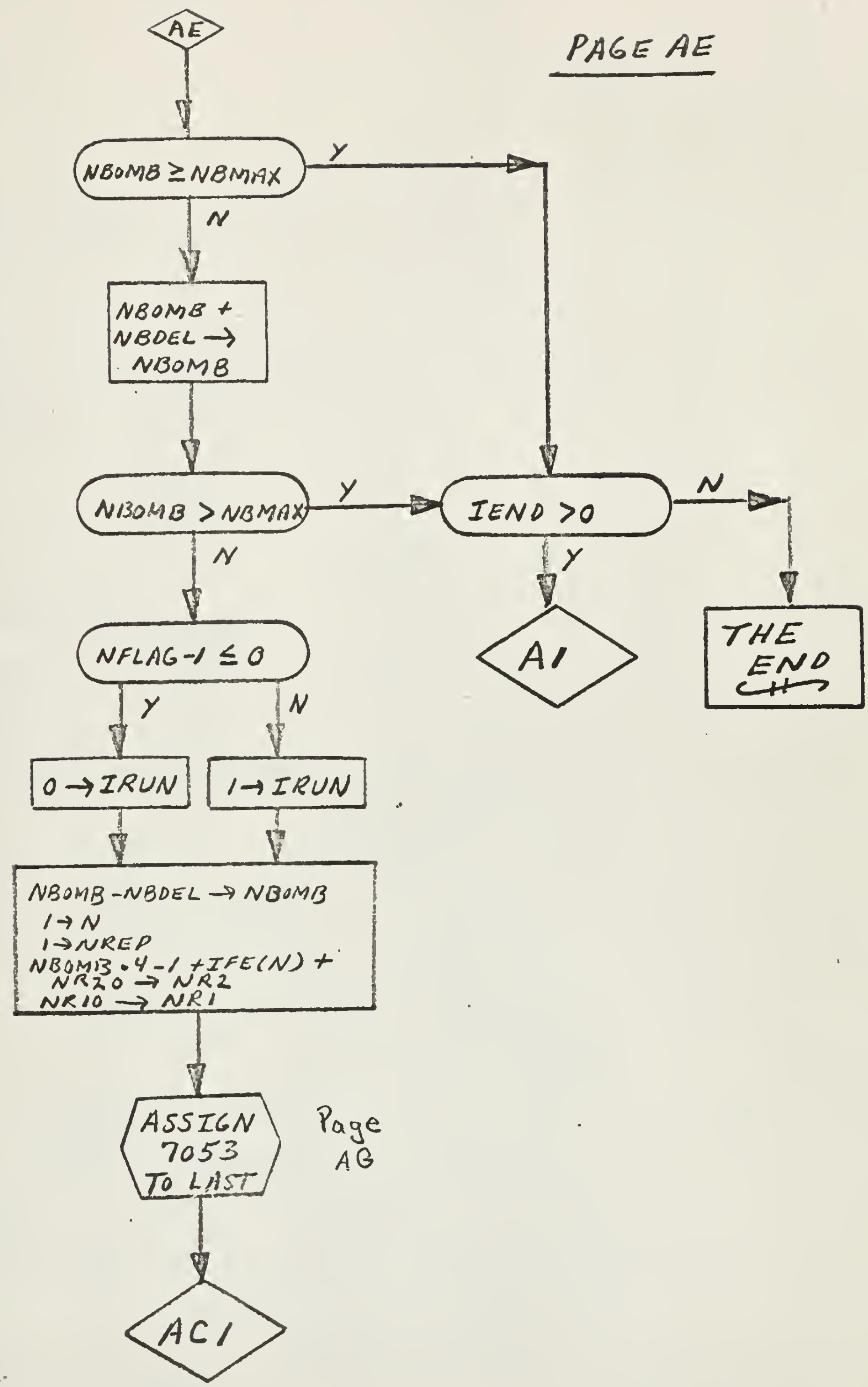
PAGE AB



AB
PRINT TITLE FOR SUMMARY UNCOORD.
AB2
PRINT SUMMARY
NFLAG-1 ≥ 0
Y
Go To REPEAT AND END
Page AC
N
>0
ERROR EXIT
K-2
=0
<0
2 → K
AB1
NFLAG-1 ≤ 0
N
START NEW PAGE
Y
AB2
PRINT TITLE FOR SUMMARY COORD.

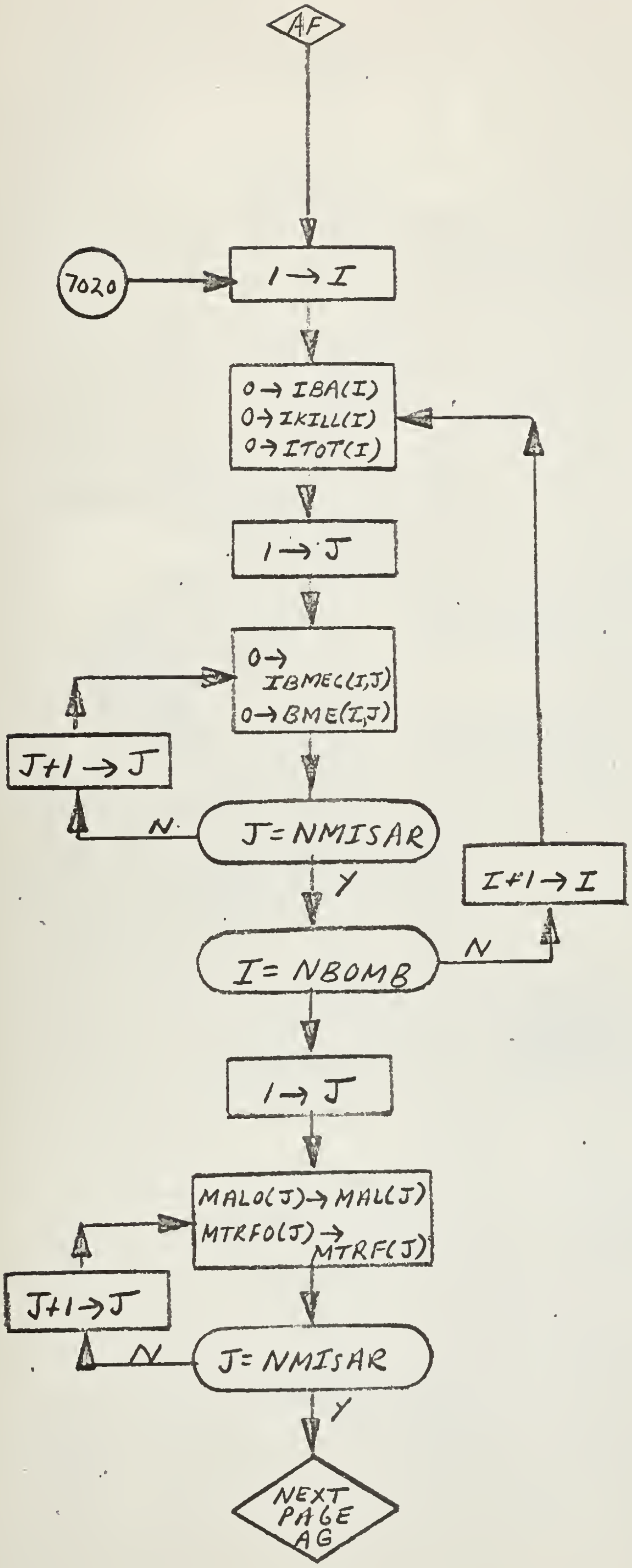

Start REPEAT AND END
0 → NOL
NFLAG-1 ≥ 0
Y
NEXT PAGE AD
N
ERROR EXIT
<0
IRUN
>0
=0
1 → IRUN
ASSIGN 7020 TO LAST
Page AF
AC1
1 → I
1 → J
BTMA(I,J,N)=0
Y
N
BTMA(I,J,N) → S1
SNE FIRE
Page S
J+1 → J
N
J = NMISAR
Y
I+1 → I
GO TO LAST
Y
I = NBOMB
N


AD
PAGE AD
N < NRPL
N
NEXT PAGE AE
Y
NREP=0
Y
N
NFLAG-1 ≤ 0
Y
N
NFLAG-1 ≤ 0
Y
0→IRUN
1→IRUN
0→IRUN
1→IRUN
NR1+NRDEL1 → NR1
N+1 → N
NBOMB-NBDEL → NBOMB
NR1+NRDEL1 → NR1
NR2+NRDEL2 → NR2
N+1 → N
ASSIGN 7041 To LAST
Page AG
PAGE AF
AC1

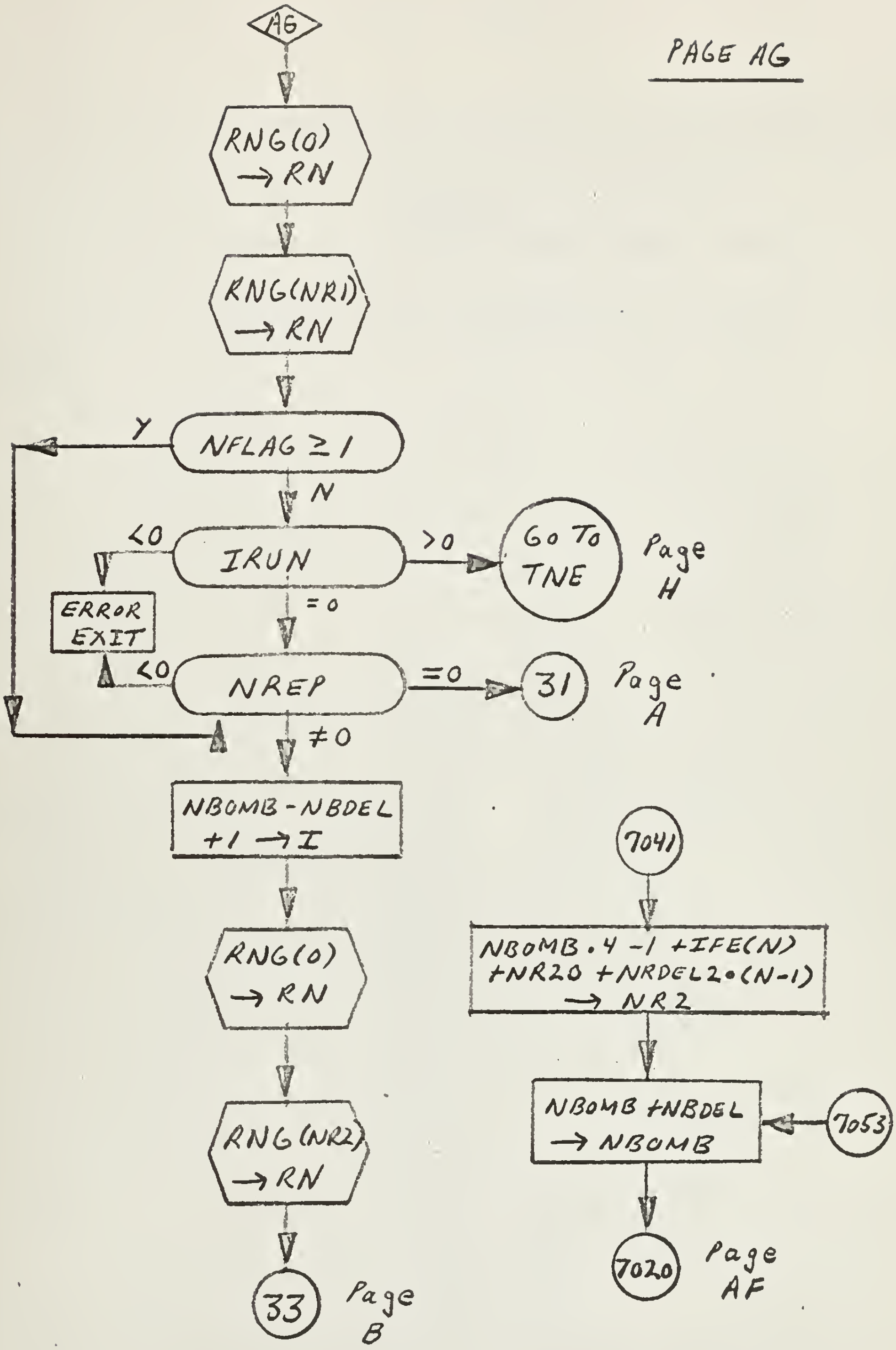


PAGE AE
AE
NBOMB ≥ NBMAX
Y
N
NBOMB + NBDEL → NBOMB
NBOMB > NBMAX
Y
N
IEND > 0
N
Y
A1
THE END
NFLAG-1 ≤ 0
Y
N
0 → IRUN
1 → IRUN
NBOMB - NBDEL → NBOMB
1 → N
1 → NREP
NBOMB · 4 - 1 + IFE(N) + NR20 → NR2
NR10 → NR1
ASSIGN 7053 To LAST
Page AG
AC1

PAGE AF


AF
7020
1 → I
0 → IBA(I)
0 → IKILL(I)
0 → ITOT(I)
1 → J
0 → IBMEC(I,J)
0 → BME(I,J)
J+1 → J
N
J = NMISAR
Y
I+1 → I
N
I = NBOMB
1 → J
MALO(J) → MAL(J)
MTRFO(J) → MTRF(J)
J+1 → J
N
J = NMISAR
Y
NEXT PAGE AG

PAGE AG


AG
RNG(0) → RN
RNG(NR1) → RN
NFLAG ≥ 1
Y
N
IRUN
<0
>0
=0
Go To TNE
Page H
ERROR EXIT
NREP
<0
=0
≠0
31
Page A
NBOMB - NBDEL +1 → I
RNG(0) → RN
RNG(NR2) → RN
33
Page B
7041
NBOMB · 4 - 1 + IFE(N) + NR20 + NRDEL2 · (N-1) → NR2
NBOMB + NBDEL → NBOMB
7053
7020
Page AF

APPENDIX II

Program Names

This appendix contains the name and definition of the symbols appearing in the computer program and the flow charts. All symbols of importance are included except those already defined in Section 7. The names in this appendix are listed in alphabetical order.

NAME	DEFINITION
ABAR(I,K)	The sample mean for the number of kills by all missile areas in replication I.*
ALT(I,N)	The altitude of aircraft I in replication N.
AM(I,J,K)	The number of kills for missile area J in replication I.*
AMT(I,K)	The total number of kills for all missile areas in replication I.*
APK(J)	The salvo kill probability for missile area J.
ASD(I,K)	The sample standard deviation for the number of kills by all missile areas in replication I.*
AVAR(I,K)	The sample variance for the number of kills by all missile areas in replication I.*
BDPC(I,J,N)	The distance from missile area J to the point of closest approach for aircraft I in replication N.
BM(J,K)	The sample mean for the number of kills by missile area J in one run.*
BME(I,J)	A non-zero value indicates that aircraft I is currently engaged by missile area J. Before intercept the value is the time of intercept and after intercept the value is the time of intercept plus assessment time. A zero value indicates no current engagement.
BMT(K)	The sample mean for the number of kills for all missile areas for one run.
BRH(I,N)	The radar horizon for aircraft I in replication N.
BS(I,N)	The speed of aircraft I in replication N.
BSD(J,K)	The sample standard deviation for the total number of kills for missile area J in one run.*
BSDT(K)	The sample standard deviation for the total number of kills for all missile areas in one run.*

NAME	DEFINITION
BT(I,N)	The time aircraft I enters the playing area in replication N.
BTMA(I,J,N)	The earliest possible time missile area J can fire at aircraft I in replication N.
BTPC(I,J,N)	The time aircraft I is at the point of closest approach for missile area J in replication N.
BV(J,K)	The sample variance for the total number of kills for missile area J in one run.*
BVT(K)	The sample variance for the total number of kills for all missile areas in one run.*
BVX(I,N)	The X-component of velocity for aircraft I in replication N.
BVY(I,N)	The Y-component of velocity for aircraft I in replication N.
BX(I,N)	The X-coordinate of the entry point into the playing area for aircraft I in replication N.
BY(I,N)	The Y-coordinate of the entry point into the playing area for aircraft I in replication N.
CBAR(I,K)	The sample variance for the number of salvos fired by all missile areas in replication I.*
CSD(I,K)	The sample standard deviation for the number of salvos fired by all missile areas in replication I.*
CTOT(I,K)	The total number of salvos fired by all missile areas in replication I.*
CTOTS(I,J,K)	The total number of salvos fired by missile area J in replication I.*
CVAR(I,K)	The sample variance for the number of salvos fired by all missile areas in replication I.*
DM(J,K)	The sample mean for the number of salvos fired by missile area J in one run.*
DMT(K)	The sample mean for the number of salvos fired by all missile areas in one run.*

NAME	DESCRIPTION
DSD(J,K)	The sample standard deviation for the number of salvos fired by missile area J in one run.*
DSDT(K)	The sample standard deviation for the total number of salvos fired by all missile areas in one run.*
DV(J,K)	The sample variance for the number of salvos fired by missile area J in one run.*
DVT(K)	The sample variance for the total number of salvos fired by all missile areas in one run.*
EVS(I)	The time of the Ith. event in the Event Store List.
IBA(I)	A non-zero value indicates that aircraft I has been killed in the current replication.
IBMEC(I,J)	The number of salvos fired at aircraft I by missile area J in the current replication.
IEVS1(I)	The type of event in the Ith. position in the Event Store List.
IEVS2(I)	The aircraft number associated with the Ith. event in the Event Store List.
IEVS3(I)	The missile area number associated with the Ith. event in the Event Store List.
IE1	The type of the current event.
IE2	The aircraft number associated with the current event.
IE3	The missile area number associated with the current event.
IFE(I)	The total number of initial FIRE events stored in replication I.
IKILL(I)	A non-zero value indicates the number of the missile area that killed aircraft number I in the current replication.
IRUN	A 0 or 1 value indicates respectively that the program is executing an uncoordinated or coordinated replication.

NAME	DESCRIPTION
IS1	The type of event being stored.
IS2	The aircraft number associated with the event being stored.
IS3	The missile area number associated with the event being stored.
ITOT(I)	The total number of salvos fired by all missile areas at aircraft I in the current replication.
MAL(J)	The current number of loaded missile launchers at missile area J
MALO(J)	The original number of loaded missile launchers at missile area J.
MTRF(J)	The current number of available tracking radars at missile area J.
MTRFO(J)	The original number of available tracking radars at missile area J.
NOL	The number of events currently on the Event Store List.
NREP	A non-zero value indicates a new set of aircraft will be generated for the next execution of the program.
NR10	The original value of NR1.
NR20	The original value of NR2.
NX	The usual argument for RNG(NX) with a value of 1.
PS(K)	The probability of survival for the Summary Output.
RN	The functional value of RNG(A), i.e., the value of the computer generated random number.
RNG(A)	The random number generating function. The value of A determines: A = 0 : generate the very first number in the sequence. A = k $\neq$ 0: generate the number k positions from the last number generated.

NAME	DESCRIPTION
S1	The time of the event being stored.
T	The time of the current event.
TF	The time of flight for the missile as calculated in the FIRE event.

NOTE: *: K = 0 or 1 indicates that the replication or run being processed is respectively uncoordinated or coordinated.

APPENDIX III

Computer Program

This appendix contains the complete computer program. The program is in FORTRAN-60 for the CDC-1604. All numerical restrictions and variable names are such that the program is compatible with IBM FORTRAN-II for the IBM-7090-94. On the CDC-1604 the program does not compile in one pass because of space allocations.

The random number generator, FUNCTION RNG(NR), appearing at the end of the program is a combination of both the assembly and compiler language for the CDC-1604. This function would have to be reprogrammed for the IBM-7090-94. The logic description of this function is as follows:

RNG is a random number generator subroutine to be used with FORTRAN on the IBM 7090 or IBM 7094. It produces a floating point random number which is uniformly distributed over the open interval (0, 1). The generator produces random numbers according to the cycle

$$X_i = X_{i-1} \times 5^{15} \pmod{2^{35}}$$

$$RN = 2^{-35} \times X_i$$

where $$X_o = 5^{15} .$$

The routine is a standard FORTRAN function - type subprogram, named RNG(NR). Its argument I is a fixed point variable or fixed point constant which tells how many times to cycle the generator from its previous value. If I = 0, the generator is reset

(X_o is set to 5^{15}) and the first number is $5^{15} \times 2^{-35}$.

The example below is given as RNG(NR) might be used.

```
1     A  =  RNG(0)

2     B  =  RNG(50)

3     C  =  RNG(1)

4     D  =  RNG(1)
```

The first statement causes the generator to be reset and $5^{15} \times 2^{-35}$ is put in A. In the second statement, the generator is cycled fifty times from its previous value and the floating point result is placed in B. The last two statements place the next two numbers generated in C and D respectively.

```
      PROGRAM REGAME
      DIMENSION IBA(20),IBMEC(20,3),BME(20,3),BVX(20,20),
     1          BVY(20,20),APK(3),MTRF(3),IKILL(20),ITOT(20),
     2          MTRFO(3),MALO(3),AM(20,3,2),AMT(20,2),
     3          ABAR(20,2),AVAR(20,2),ASD(20,2),BM(3,2),
     4          BMT(2),BV(3,2),BVT(2),BSD(3,2),BSDT(2),PS(2),
     5          EVS(200),IEVS1(200),IEVS2(200),IEVS3(200),
     6          AX(3),AY(3),MAL(3),RMR(3),RMAX(3),AVS(3),
     7          ATM(3),ATX(3),ASM(3),ASX(3),RTM(3)
      DIMENSION RTX(3),PK(3),BX(20,20),BY(20,20),BS(20,20),
     1          BT(20,20),BRH(20,20),ALT(20,20),BTMA(20,3,20),
     2          BTPC(20,3,20),BDPC(20,3,20),CTOTS(20,3,2),
     3          CTOT(20,2),CBAR(20,2),CVAR(20,2),CSD(20,2),
     4          DM(3,2),DMT(2),DV(3,2),DVT(2),DSD(3,2),
     5          DSDT(2),NAME(2),IFE(20)
C
C     READ INPUT
C
    1 READ INPUT TAPE 5,2    ,NMISAR,(MTRF(I),I=1,3),(MAL(I),
     1       I=1,3),(AX(I),I=1,3),(AY(I),I=1,3),(RMR(I),I=1,3)
     2       ,(RMAX(I),I=1,3),(AVS(I),I=1,3),IEND,NAME(1),NAME(2)
    2 FORMAT (I1,3I2,3I1,12F3.0,3F4.0,I1,A8,A5)
      READ INPUT TAPE 5,3    ,(PK(I),I=1,3),(ATM(I),I=1,3),
     1       (ATX(I),I=1,3),(ASM(I),I=1,3),(ASX(I),I=1,3),
     2       (RTM(I),I=1,3),(RTX(I),I=1,3)
    3 FORMAT (3F2.2,18F3.0)
      READ INPUT TAPE 5,4    ,NBMAX,BSX,BSM,BTX,BTM,NBOMB,
     1       NBDEL,GX,GY,RCC,CPA,TMAX,OUTPUT,NR1,NRDEL1,
     2       NR2,NRDEL2,NRPL,AMAX,AMIN
    4 FORMAT (I3,2F4.0,2F3.0,2I3,4F3.0,F2.0,I1,4I3,I2,3P2F3.0)
      READ INPUT TAPE 5, 494, STEP, NFLAG, IHIST
  494 FORMAT (F3.0, 2I2)
      PRINT 9991, NAME(1),NAME(2)
 9991 FORMAT (3X,2A8)
C
C     SET CONSTANTS
C
      IF (NFLAG-1) 495,495,496
  495 IRUN=0
      GO TO 497
  496 IRUN=1
  497 NRRN=0
      N=1
      NREP=0
      NOL  = 0
      DO 5 I=1,20
      IKILL(I) =0
      IBA(I)   =0
      ITOT(I)  =0
      IFE(I)=0
      DO 5 J=1,3
      BME(I,J)=0.
      IBMEC(I,J)=0
    5 CONTINUE
      IF(NMISAR-3) 6,8,9999
```

```
 6 L=NMISAR+1
   DO 7 K=1,2
   DO 7 J=L,3
   BM(J,K)=0.
   BV(J,K)=0.
   BSD(J,K)=0.
   DM(J,K)=0.
   DV(J,K)=0.
   DSD(J,K)=0.
   DO 7 I=1,NRPL
   AM(I,J,K)=0.
   CTOTS(I,J,K)=0.
 7 CONTINUE
C
C  PRINT INPUT
C
 8 WRITE OUTPUT TAPE 6, 9   ,NAME(1),NAME(2)
 9 FORMAT (11H1INPUT DATA,75X,2A8///)
   WRITE OUTPUT TAPE 6,10  , GX,GY,RCC,CPA
10 FORMAT(5X,15HGAME PARAMETERS//, 10X,
 1       30H XCOORDINATE OF CENTER (GX) = F8.2,2X,5HMILES/10X,
 2       30H YCOORDINATE OF CENTER (GY) = F8.2,2X,5HMILES/10X,
 3       31H RADIUS OF CIRCLE      (RCC) = F7.2,2X,5HMILES/10X,
 4       31H CENTRAL PLAYING ANGLE (CPA) = F7.2,2X, 7HDEGREES////)
C
   WRITE OUTPUT TAPE 6, 11
11 FORMAT(25H MISSILE AREA INFORMATION//1X, 16H MISSILE AREA NR
 1         8X, 1H1, 9X, 1H2, 9X, 1H3/)
   WRITE OUTPUT TAPE 6, 12  , (AX(I), I=1,3)
12 FORMAT (21X, 3(F6.2,4X), 5HMILES, 7X,19HAX    = XCOORDINATE)
   WRITE OUTPUT TAPE 6,13  , (AY(I),I=1,3)
13 FORMAT(21X, 3(F6.2,4X), 5HMILES, 7X,19HAY    = YCOORDINATE)
   WRITE OUTPUT TAPE 6, 14 , (MAL(I),I=1,3)
14 FORMAT(21X, I3,7X, I3, 7X, I3, 19X,  23HMAL   = NR OF LAUNCHERS)
   WRITE OUTPUT TAPE 6, 15 , (MTRF(I),I=1,3)
15 FORMAT (21X,I3,7X,I3,7X,I3,19X,22HMTRF  = NR OF T RADARS)
   WRITE OUTPUT TAPE 6, 16 , (RMR(I),I=1,3)
16 FORMAT(21X, 3(F6.2,4X), 5HMILES, 7X, 30HRMR   = SEARCH RADAR MAX R
  1ANGE)
   WRITE OUTPUT TAPE 6, 17 , (RMAX(I),I=1,3)
17 FORMAT(21X, 3(F6.2,4X), 5HMILES, 7X,25HRMAX  = MISSILE MAX RANGE)
   WRITE OUTPUT TAPE 6, 18 , (AVS(I),I=1,3)
18 FORMAT(20X, 3(F7.2,3X),1X, 5HMI/HR,7X,25HAVS   = MISSILE AVG SPEED
  *)
   WRITE OUTPUT TAPE 6, 19 , (ATM(I),I=1,3)
19 FORMAT(21X, 3(F6.2,4X), 5HMINS , 7X,32HATM   = ACQUISITION TIME MI
  1NIMUM)
   WRITE OUTPUT TAPE 6, 20 , (ATX(I),I=1,3)
20 FORMAT(21X, 3(F6.2,4X), 5HMINS , 7X,32HATX   = ACQUISITION TIME MA
  1XIMUM)
   WRITE OUTPUT TAPE 6,21  , (ASM(I),I=1,3)
21 FORMAT(21X, 3(F6.2,4X), 5HMINS , 7X,31HASM   = ASSESSMENT TIME MIN
  1IMUM)
   WRITE OUTPUT TAPE 6, 22 , (ASX(I),I=1,3)
22 FORMAT(21X, 3(F6.2,4X), 5HMINS , 7X,31HASX   = ASSESSMENT TIME MAX
  1IMUM)
```

```
      WRITE OUTPUT TAPE 6,  23 , (RTM(I), I=1,3)
   23 FORMAT(21X, 3(F6.2,4X), 5HMINS , 7X,27HRTM    = RELOAD TIME MINIMUM
     1)
      WRITE OUTPUT TAPE 6,  24 , (RTX(I),I=1,3)
   24 FORMAT(21X, 3(F6.2,4X), 5HMINS , 7X,27HRTX    = RELOAD TIME MAXIMUM
     1)
      DO  25  I=1,3
   25 APK(I)=100.* PK(I)
      WRITE OUTPUT TAPE 6, 26  , (APK(I), I=1,3)
   26 FORMAT(21X, 3(F6.2,4X), 5HPER. , 7X,34HPK    = KILL PROBABILITY PE
     1R SALVO)
      WRITE OUTPUT TAPE 6, 27
   27 FORMAT(/// 18HBOMBER INFORMATION//)
      WRITE OUTPUT TAPE 6,  28 , BSX,BSM,AMAX,AMIN,BTX,BTM
   28 FORMAT(5X,21H MAXIMUM SPEED (BSX)= F6.1,2X, 8HMILES/HR/
     1       5X,21H MINIMUM SPEED (BSM)= F6.1,2X, 8HMILES/HR/
     2       5X,26H MAXIMUM ALTITUDE (AMAX)= F8.1,2X,4HFEET/
     3       5X,26H MINIMUM ALTITUDE (AMIN)= F8.1,2X,4HFEET/
     4       5X,31H MAXIMUM SEPARATION TIME (BTX)= F7.2,2X, 5HMINS./
     5        5X,31H MINIMUM SEPARATION TIME (BTM)= F7.2,2X,5HMINS./1H1)
C
C     INITIALIZE
C
      DO 29 I=1,NMISAR
      ATM(I)=ATM(I)/60.0
      ATX(I)=ATX(I)/60.0
      ASM(I)=ASM(I)/60.0
      ASX(I)=ASX(I)/60.0
      RTM(I)=RTM(I)/60.0
   29 RTX(I)=RTX(I)/60.0
      CPA=CPA/57.29577
      BTX=BTX/60.0
      BTM=BTM/60.0
      STEP=STEP/60.0
      NX=1
      DO 30 J=1,3
      MALO(J)=MAL(J)
   30 MTRFO(J)=MTRF(J)
      NR10=NR1
      NR20=NR2
C
C     COMPUTE BOMBER ENTRY POINTS
C
   31 RN=RNG(0)
      RN=RNG(NR2)
   32 I=1
   33 RN=RNG(NX)
      IF (RN-0.5) 34,36,35
   34 B=CPA*(0.5-RN)
      X=RCC*COSF(B)
      Y=RCC*SINF(B)
      BX(I,N)=GX+X
      BY(I,N)=GY+Y
      K=1
      GO TO 37
   35 B=CPA*(RN-0.5)
```

```
      X=RCC*COSF(B)
      Y=RCC*SINF(B)
      BX(I,N)=GX+X
      BY(I,N)=GY-Y
      K=2
      GO TO 37
   36 BX(I,N)=GX+RCC
      BY(I,N)=GY
      K=3
   37 RN=RNG(NX)
C
C     COMPUTE BOMBER SPEED AT ENTRY
C
      BS(I,N)=(BSX-BSM)*RN+BSM
      IF(K-1)39,38,39
   38 BVX(I,N)=-BS(I,N)*COSF(B)
      BVY(I,N)=-BS(I,N)*SINF(B)
      GO TO 42
   39 IF (K-2) 41,40,41
   40 BVX(I,N)=-BS(I,N)*COSF(B)
      BVY(I,N)=BS(I,N)*SINF(B)
      GO TO 42
   41 BVX(I,N)=-BS(I,N)
      BVY(I,N)=0.
C
C     COMPUTE BOMBER TIME AT ENTRY
C
   42 IF(I-1) 44,43,44
   43 BT(I,N)=1.0
      GO TO 45
   44 RN=RNG(NX)
      BT(I,N)=(BTX-BTM)*RN+BTM+BT(I-1,N)
C
C     COMPUTE BOMBER ALTITUDE AND BOMBER RADAR HORIZON
C
   45 RN=RNG(NX)
      ALT(I,N)=(AMAX-AMIN)*RN+AMIN
      BRH(I,N)=1.25*SQRTF(ALT(I,N))
      ALT(I,N)=ALT(I,N)/5280.0
   46 J=1
   47 A2=(BX(I,N)-AX(J))**2+(BY(I,N)-AY(J))**2
      A=SQRTF(A2)
      B2=(AX(J)-GX)**2 + (AY(J)-GY)**2
      B=SQRTF(B2)
      C2= RCC*RCC
      C=RCC
      CB= (-B2+A2+C2)/(2.*A*C)
      SB2= 1.0-CB**2
      SB=SQRTF(SB2)
C
C     COMPUTE DISTANCE TO PT OF CLOSEST APPROACH FROM MISSILE AREA
C
      BDPC(I,J,N)=A*SB
      IF(BDPC(I,J,N)-RMAX(J))48,48,57
   48 IF(A-RMR(J)) 55,55,49
   49 IF(BDPC(I,J,N)-RMR(J))50,50,57
```

```
   50 IF(BRH(I,N)-RMR(J))505,51,51
   51 IF (RMR(J)-RMAX(J)) 52,52,507
   52 Z2=RMR(J)**2-BDPC(I,J,N)**2
      GO TO 53
  505 IF(BRH(I,N)-BDPC(I,J,N))57,506,506
  506 IF (BRH(I,N)-RMAX(J)) 508,508,507
  507 X=A*CB
      BTPC(I,J,N)= X/BS(I,N)+BT(I,N)
      Z=BDPC(I,J,N)**2
      ZR2=RMR(J)**2 - Z
      ZM2= RMAX(J)**2 - Z
      ZB2= BRH(I,N)**2 - Z
      ZM = SQRTF(ZM2)
      IF (ZR2 - ZB2) 800, 801, 801
  800 ZR= SQRTF(ZR2)
      GO TO 802
  801 ZR= SQRTF(ZB2)
  802 TRMAX= (X-ZM)/BS(I,N) + BT(I,N)
      TZR= (X-ZR)/BS(I,N) + BT(I,N)
      Z= RMAX(J)**2 + ALT(I,N)**2
      S1= TRMAX - SQRTF(Z)/AVS(J)
      RN=RNG(NX)
      Z = (ATX(J)-ATM(J))*RN + ATM(J)
       IF ((S1-Z)-TZR) 804,803,803
  803 BTMA(I,J,N) = S1
      GO TO 56
  804 S1= TZR+Z
      GO TO 803
  508 Z2=BRH(I,N)**2-BDPC(I,J,N)**2
   53 Z=SQRTF(Z2)
      X= A*CB
C
C     COMPUTE TIME AT PT OF CLOSEST APPROACH
C
      BTPC(I,J,N)=X/BS(I,N)+BT(I,N)
      Y=X-Z
C
C     TIME MISSILE AREA ENTRY
C
      BTMA(I,J,N)=BT(I,N)+Y/BS(I,N)
   54 RN= RNG(NX)
      S1=BTMA(I,J,N)+(ATX(J)-ATM(J))*RN+ATM(J)
      BTMA(I,J,N)=S1
      GO TO 56
   55 IF (A-BRH(I,N)) 555,555,49
  555 BTMA (I,J,N) = BT(I,N)
      X=SQRTF(A**2-BDPC(I,J,N)**2)
      BTPC(I,J,N)=X/BS(I,N)+BT(I,N)
      GO TO 54
   56 IS1=5
      IS2=I
      IS3=J
      IFE(N)=IFE(N)+1
      ASSIGN 58 TO NEXT
      GO TO 700
   57 BTMA(I,J,N)=0.
```

```
 58 IF(J-NMISAR) 59,60,9999
 59 J=J+1
    GO TO 47
 60 IF(I-NBOMB) 61,62,9999
 61 I=I+1
    GO TO 33
 62 RN=RNG(0)
    RN=RNG(NR1)
    GO TO 600
C
C   TNE
C
600 IF(NOL) 9999,6000,601
601 T=EVS(1)
    IE1=IEVS1(1)
    IE2=IEVS2(1)
    IE3=IEVS3(1)
    IF (IHIST) 9999, 952, 950
950 WRITE OUTPUT TAPE 6, 951, T, IE1, IE2, IE3
951 FORMAT (10X, F7.3, 3I5)
952 IF (NOL-1) 9999, 604, 602
602 DO 603 I=2,NOL
    EVS(I-1) = EVS(I)
    IEVS1(I-1) = IEVS1(I)
    IEVS2(I-1) = IEVS2(I)
603 IEVS3(I-1) = IEVS3(I)
604 EVS(NOL) =0.
    IEVS1(NOL) = 0
    IEVS2(NOL) = 0
    IEVS3(NOL) = 0
    NOL = NOL - 1
    IF ( T - TMAX )  605,605,608
605 IF (IE1 ) 9999,9999,606
606 IF (IE1 - 5)  607,607,9999
607 GO TO (1000,2000,3000,4000,5000),IE1
608 WRITE OUTPUT TAPE 6,609, NBOMB,NRPLI
609 FORMAT (26H1MAXIMUM TIME EXCEEDED FOR I5, 8HBOMBERS ,
  1          4HRUN I3)
    GO TO 6000
C
C   SNE
C
700 IF (NOL) 9999, 701, 702
701 M = 1
    GO TO 706
702 M  = NOL
703 IF(S1 - EVS(M)) 704,709,708
704 EVS(M+1) = EVS(M)
    IEVS1(M+1) = IEVS1(M)
    IEVS2(M+1) = IEVS2(M)
    IEVS3(M+1) = IEVS3(M)
    IF ( M-1 )9999,706,705
705 M = M-1
    GO TO 703
706 IF (IHIST) 9999, 955, 953
953 WRITE OUTPUT TAPE 6, 954, S1, IS1, IS2, IS3
```

```
  954 FORMAT (20X, F7.3, 3I5)
  955 EVS(M)=S1
      IEVS1(M) = IS1
      IEVS2(M) = IS2
      IEVS3(M) = IS3
      NOL = NOL + 1
  707 GO TO NEXT
  708 M = M + 1
      GO TO 706
  709 IF(IS1 - IEVS1(M))  704,710,708
  710 IF(IS2 - IEVS2(M))  704,711,708
  711 IF(IS3 - IEVS3(M))  704,707,708
C
C     RELOAD
C
 1000 J = IE3
      MAL(J) = MAL(J) + 1
      GO TO 600
C
C     T R FREE
C
 2000 J = IE3
      MTRF(J) = MTRF(J) + 1
      GO TO 600
C
C     INTERCEPT
C
 3000 I=IE2
      J=IE3
      IF (IBA(I)) 600,3001,600
 3001 RN=RNG(NX)
      IF(RN-PK(J)) 3002,3002,3009
 3002 IF(IRUN) 3013,3012,3013
 3012 K=1
 3003 IF(BME(I,K))3004,3006,3004
 3004 RN=RNG(NX)
      S1=T+(ASX(K)-ASM(K)) *RN+ASM(K)
      BME(I,K)=S1
      IS3=K
      IS1=2
      IS2=I
      ASSIGN 3005 TO NEXT
      GO TO 700
 3005 IS1=4
      IS2=I
      ASSIGN 3006 TO NEXT
      GO TO 700
 3006 IF(K-NMISAR)3007,3008,3007
 3007 K=K+1
      GO TO 3003
 3008 IBA(I)=1
      IKILL(I) = J
      GO TO 600
 3009 RN=RNG(NX)
      S1=T+(ASX(J)-ASM(J))*RN+ASM(J)
      BME(I,J)=S1
```

```
      IS1=5
      IS2=I
      IS3=J
      ASSIGN 3010 TO NEXT
      GO TO 700
 3010 IS1=2
      IS2=I
      ASSIGN 3011 TO NEXT
      GO TO 700
 3011 IS1=4
      IS2=I
      ASSIGN 600 TO NEXT
      GO TO 700
 3013 RN=RNG(NX)
      S1=T+(ASX(J)-ASM(J))*RN+ASM(J)
      BME(I,J)=S1
      IS1=2
      IS2=I
      IS3=J
      ASSIGN 3014 TO NEXT
      GO TO 700
 3014 IS1=4
      ASSIGN 3008 TO NEXT
      GO TO 700
C
C     ENGAGEMENT
C
 4000 I = IE2
      J = IE3
      BME(I,J)=0.
      GO TO 600
C
C     FIRE EVENT
C
 5000 I=IE2
      J=IE3
      IF(T-BTPC(I,J,N))5001,600,600
 5001 IF(IBA(I)) 600, 5002, 600
 5002 IF(MAL(J)) 5003, 5010, 5003
 5003 IF (MTRF(J)) 5004, 5012, 5004
 5004 IF(IRUN) 5005,5013,5005
 5005 K=1
 5006 IF(BME(I,K)) 5020,5007,5020
 5020 S1=BME(I,K)
      GO TO 5011
 5007 IF(K-NMISAR) 5008,5013,5008
 5008 K=K+1
      GO TO 5006
 5010 S1=T+STEP
 5011 IS1=5
      IS2=I
      IS3=J
      ASSIGN 600 TO NEXT
      GO TO 700
 5012 S1=T+STEP
      GO TO 5011
```

```
5013 XA=BX(I,N)+BVX(I,N)*(T-BT(I,N))
     YA=BY(I,N)+BVY(I,N)*(T-BT(I,N))
     A=BS(I,N)**2 - AVS(J)**2
     B= 2.0 * (BVX(I,N)*(XA-AX(J))+BVY(I,N)*(YA-AY(J)))
     C= (XA-AX(J))**2 + (YA-AY(J))**2+ ALT(I,N)**2
     IF (A) 5114, 5123, 5114
5114 Z= B**2-4.0*A*C
     IF (Z) 600, 5115, 5115
5115 TF1= (-B + SQRTF(Z))/(2.0*A)
     TF2= (-B-SQRTF(Z))/(2.0*A)
     IF (TF1) 5116,600,5116
5116 IF (TF2) 5117,600,5117
5117 IF (TF1) 5120,5118,5118
5118 IF (TF2) 5119,5122,5122
5119 TF=TF1
      GO TO 5015
5120 IF (TF2) 600, 5121, 5121
5121 TF=TF2
     GO TO 5015
5122 IF (TF1-TF2) 5119, 5121, 5121
5123 TF = -C/B
     IF (TF)600,600,5015
5015 R= (TF * AVS(J))**2 - ALT(I,N)**2
     R= SQRTF(R)
     IF ((R-1.)-RMAX(J)) 5016,5016,5019
5016 S1=T+TF
     IS1=3
     IS2=I
     IS3=J
     ASSIGN 5017 TO NEXT
     GO TO 700
5017 MAL(J)=MAL(J)-1
     MTRF(J)=MTRF(J)-1
     IBMEC(I,J)=IBMEC(I,J)+1
     BME(I,J)=S1
     RN=RNG(NX)
     S1=T+(RTX(J)-RTM(J))*RN+RTM(J)
     IS1=1
     IS2=I
     IS3=J
5018 ASSIGN 600 TO NEXT
     GO TO 700
5019 S1=T+STEP
     IS1=5
     IS2=I
     IS3=J
     GO TO 5018
C
C    OUTPUT
C
6000 IF( OUTPUT) 9999, 6001, 6015
6001 DO 6002 I=1,NBOMB
     ITOT(I) = 0
     DO 6002 J=1,NMISAR
6002 ITOT(I)= ITOT(I) + IBMEC(I,J)
     IF (NFLAG-1) 6003,6004,6006
```

```
 6003 IF(IRUN) 9999,6004,6006
 6004 WRITE OUTPUT TAPE 6,6005,N
 6005 FORMAT(/////,35H UNCOORDINATED ATTACK RESULTS   REP,I3,//)
      GO TO 6008
 6006 WRITE OUTPUT TAPE 6,6007,N
 6007 FORMAT(/////,33H COORDINATED ATTACK RESULTS   REP,I3,//)
 6008 WRITE OUTPUT TAPE 6, 6009
 6009 FORMAT ( 45H BMR SMA1 SMA2 SMA3 TOTAL A/K XCOORD YCOORD
     1  51HBRH  ALTITUDE SPEED  TIME  TIME(1) TIME(2) TIME(3) //)
      DO 6010 I=1,NBOMB
 6010 WRITE OUTPUT TAPE 6,6011,I, (IBMEC(I,J),J=1,3),ITOT(I),
     1     IKILL(I),BX(I,N),BY(I,N),BRH(I,N),ALT(I,N),BS(I,N),
     2   BT(I,N),(BTMA(I,J,N),J=1,NMISAR)
 6011 FORMAT (I4, 3I5, I6, I4, 3F7.2, F7.2, F8.2, F6.3, F7.3,2F8.3)
      WRITE OUTPUT TAPE 6, 60115
60115 FORMAT (///)
      GO TO 6015
 6015 I=N
      IF( IRUN ) 9999,6016,6017
 6016 K=1
      GO TO 6018
 6017 K=2
 6018 DO 6019 J=1,NMISAR
      CTOTS(I,J,K)=0
      AM(I,J,K)=0
      DO 6019 L=1,NBOMB
      B=IBMEC(L,J)
      CTOTS(I,J,K) =CTOTS(I,J,K) +B
 6019 CONTINUE
      DO 6021 L=1,NBOMB
      IF(IKILL(L)) 9999,6021,6020
 6020 J=IKILL(L)
      AM(I,J,K)=AM(I,J,K)+1.0
 6021 CONTINUE
      CTOT(I,K) = 0.
      AMT(I,K) =0.
      DO 6022 J=1,NMISAR
      CTOT(I,K)=CTOT(I,K)+CTOTS(I,J,K)
 6022 AMT(I,K)=AMT(I,K)+AM(I,J,K)
      A=NMISAR
      ABAR(I,K)=AMT(I,K)/A
      CBAR(I,K)=CTOT(I,K)/A
      CVAR(I,K) =0.
      AVAR(I,K) =0.
      DO 6023 J=1,NMISAR
      CVAR(I,K)=CVAR(I,K)+(CTOTS(I,J,K)-CBAR(I,K))**2
 6023 AVAR(I,K)=AVAR(I,K)+(AM(I,J,K)-ABAR(I,K))**2
      AVAR(I,K)=AVAR(I,K)/A
      CVAR(I,K)=CVAR(I,K)/A
      CSD(I,K)=SQRTF(CVAR(I,K))
      ASD(I,K)=SQRTF(AVAR(I,K))
      IF(N-NRPL) 7000,6024,7000
 6024 IF (NFLAG-1) 60245,6025,6035
60245 IF (IRUN) 6025,7000,6025
 6025 K=1
60255 A=NRPL
```

```
      DO 6027 J=1,NMISAR
      DM(J,K)=0
      BM(J,K)=0
      DO 6026 I=1,NRPL
      DM(J,K)=DM(J,K)+CTOTS(I,J,K)
 6026 BM(J,K)=BM(J,K)+AM(I,J,K)
      DM(J,K)=DM(J,K)/A
 6027 BM(J,K)=BM(J,K)/A
      DMT(K)=0
      BMT(K)=0
      DO 6028 I=1,NRPL
      DMT(K)=DMT(K)+CTOT(I,K)
 6028 BMT(K)=BMT(K)+AMT(I,K)
      DMT(K)=DMT(K)/A
      BMT(K)=BMT(K)/A
      DO 6030 J=1,NMISAR
      DV(J,K)=0
      BV(J,K) = 0
      DO 6029 I=1,NRPL
      DV(J,K)=DV(J,K)+(CTOTS(I,J,K)-DM(J,K))**2
 6029 BV(J,K)=BV(J,K)+(AM(I,J,K)-BM(J,K))**2
      DV(J,K)=DV(J,K)/A
 6030 BV(J,K)=BV(J,K)/A
      DVT(K)=0
      BVT(K)=0
      DO 6031 I=1,NRPL
      DVT(K)=DVT(K)+(CTOT(I,K)-DMT(K))**2
 6031 BVT(K)=BVT(K)+(AMT(I,K)-BMT(K))**2
      DVT(K)=DVT(K)/A
      BVT(K)=BVT(K)/A
      DO 6032 J=1,NMISAR
      DSD(J,K)=SQRTF(DV(J,K))
 6032 BSD(J,K)=SQRTF(BV(J,K))
      DSDT(K)=SQRTF(DVT(K))
      BSDT(K)=SQRTF(BVT(K))
      A=NBOMB
      PS(K) =0.
      DO 6034 I=1,NRPL
      IF(AMT(I,K)-A)6034,6033,6034
 6033 PS(K)=PS(K)+1.0
 6034 CONTINUE
      A=NRPL
      PS(K)=PS(K)/A
      IF (NFLAG-1) 60345,6036,6049
60345 IF (K-2) 6035,6036,6035
 6035 K=2
      GO TO 60255
 6036 K=1
 6037 WRITE OUTPUT TAPE 6,6038
 6038 FORMAT(31H1SUMMARY UNCOORDINATED RESULTS //,
     1          19X,5HKILLS,51X,11HSHOTS FIRED /)
      GO TO 6041
 6039 IF (NFLAG-1) 60395, 60395, 60390
60390 WRITE OUTPUT TAPE 6, 60405
60405 FORMAT (1H1)
60395 WRITE OUTPUT TAPE 6, 6040
```

```
 6040 FORMAT (29H SUMMARY COORDINATED RESULTS //,
    1         19X,5HKILLS,51X,11HSHOTS FIRED /)
 6041 WRITE OUTPUT TAPE 6,6042
 6042 FORMAT (34H NBOMB REP MA1 MA2 MA3 TOTAL  MEAN,4X,
    1          9HVAR    SD,7X,20HMA1  MA2  MA3  TOTAL,
    2         19H  MEAN    VAR    SD/)
      WRITE OUTPUT TAPE 6,6043, (NBOMB,I,(AM(I,J,K),J=1,3),
    1         AMT(I,K),ABAR(I,K),AVAR(I,K),ASD(I,K),
    2         (CTOTS(I,J,K),J=1,3),CTOT(I,K),CBAR(I,K),
    3         CVAR(I,K),CSD(I,K),I=1,NRPL)
 6043 FORMAT (I6,I4,3F4.0,F6.0,3F7.2,F8.0,2F5.0,F7.0,3F7.2)
      WRITE OUTPUT TAPE 6, 6044
 6044 FORMAT(//////,9X,26HMA1    MA2    MA3    TOTAL,28X,
    1        26HMA1    MA2    MA3    TOTAL/)
      WRITE OUTPUT TAPE 6,6045, (BM(J,K), J=1,3),  BMT(K),
    1         (DM(J,K),J=1,3),DMT(K)
 6045 FORMAT(6H MEAN 3F7.2,F8.2,24X,3F7.2,F8.2)
      WRITE OUTPUT TAPE 6,6046, (BV(J,K), J=1,3), BVT(K),
    1         (DV(J,K),J=1,3),DVT(K)
 6046 FORMAT (6H VAR  3F7.2,F8.2,24X,3F7.2,F8.2)
      WRITE OUTPUT TAPE 6,6047, (BSD(J,K),J=1,3), BSDT(K),
    1         (DSD(J,K),J=1,3), DSDT(K)
 6047 FORMAT(6H SD   3F7.2,F8.2,24X,3F7.2,F8.2)
      WRITE OUTPUT TAPE 6, 6048,PS(K)
 6048 FORMAT (///, 26H PROBABLITY OF SURVIVAL = F7.2,/,1H1)
      IF (NFLAG-1) 60485,7000,7000
60485 IF (K-2) 6049,7000,9999
 6049 K=2
      GO TO 6039
C
C     REPEAT AND END
C
 7000 NOL=0
      IF (NFLAG-1) 900,7030,7030
  900 IF (IRUN) 9999,7001,7030
 7001 IRUN=1
      ASSIGN 7020 TO LAST
 7005 I=1
 7006 J=1
 7007 IF(BTMA(I,J,N))7008,7009,7008
 7008 S1=BTMA(I,J,N)
      IS1=5
      IS2=I
      IS3=J
      ASSIGN 7009 TO NEXT
      GO TO 700
 7009 IF(J-NMISAR) 7010,7011,9999
 7010 J=J+1
      GO TO 7007
 7011 IF(I-NBOMB)7012,7013,9999
 7012 I=I+1
      GO TO 7006
 7013 GO TO LAST
 7020 DO 7021 I=1,NBOMB
      IBA(I)=0
      IKILL(I)=0
```

```
      ITOT(I)=0
      DO 7021 J=1,NMISAR
      IBMEC(I,J)=0
      BME(I,J)=0
 7021 CONTINUE
      DO 7022 J=1,NMISAR
      MAL(J)=MALO(J)
 7022 MTRF(J)=MTRFO(J)
      RN=RNG(0)
      RN= RNG(NR1)
      IF (NFLAG-1) 70225,70225,7060
70225 IF (IRUN) 9999,7060,600
 7030 IF(N-NRPL)7031,7050,9999
 7031 IF(NREP) 9999,7032,7040
 7032 IF (NFLAG-1) 70325,70325,810
  810 IRUN=1
      GO TO 70326
70325 IRUN=0
70326 NR1=NR1+NRDEL1
      NR2=NR2+NRDEL2
      N=N+1
      GO TO 7020
 7040 IF (NFLAG-1) 70405,70405,820
  820 IRUN=1
      GO TO 70406
70405 IRUN=0
70406 NR1=NR1+NRDEL1
      N=N+1
      NBOMB=NBOMB-NBDEL
      ASSIGN 7041 TO LAST
      GO TO 7005
 7041 NR2=NBOMB*4-1+IFE(N)+NR20+NRDEL2*(N-1)
      GO TO 7053
 7050 IF(NBOMB-NBMAX)7051,9888,9888
 7051 NBOMB=NBOMB+NBDEL
      IF(NBOMB-NBMAX)7052,7052,9888
 7052 NBOMB=NBOMB-NBDEL
      IF (NFLAG-1) 70525,70525,830
  830 IRUN=1
      GO TO 70526
70525 IRUN=0
70526 N=1
      NREP=1
      NR2=NBOMB*4-1+IFE(N)+NR20
      NR1=NR10
      ASSIGN 7053 TO LAST
      GO TO 7005
 7053 NBOMB=NBOMB+NBDEL
      GO TO 7020
 7060 IF(NREP)9999,31,7061
 7061 I=NBOMB-NBDEL+1
      RN= RNG(0)
      RN= RNG(NR2)
      GO TO 33
 9999 PRINT 9997
 9997 FORMAT ( 6H ERROR)
```

```
9888 IF(IEND)9994,9994,1
9994 CONTINUE
     END
     FUNCTION RNG(NR)
     CON (K=30517578125,
    1      K1=30517578125,
    2      K2=200000000000000B,
    3      K3=777740000000000B)
     IF (NR) 10,10,20
  10 K1 = K
     GO TO 40
  20 DO 30 I=1,NR
  30 LDA(K),MUI(K1),SCL(K3),STA(K1)
  40 ALS(1),ADD(K2),FAD(K2)
     RETURN
     END
     END
```

APPENDIX IV

Using The Program at The U. S. Naval Postgraduate School

For the FORTRAN-60 Compiler and the CDC-1604 now in use at the U. S. Naval Postgraduate School the program is included on the library tape. The program's name is REGAME and using the program requires only that the user submit a normal FORTRAN-60 job consisting of a simple FORTRAN-60 program, by which the program REGAME is loaded into memory by the statement CALL REGAME, and the necessary sets of 4 input cards as described in the Input Section.

Except for the values to be placed on the input cards the next page of this appendix is an example of the FORTRAN-60 program needed for using the REGAME program.

GRAMMER ANDRUS A.F.

PROGRAM

GE 1 OF 1

SPECIAL INSTRUCTIONS

D PUNCH OPERATOR

DATE COMPLETED

STATE-MENT NO.	CONT.	FORTRAN STATEMEN
		O = ZERO Ø = ALPHA O 1 = ONE I = ALPHA I

```
2 3 4 5 |6| 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 ... 45
 .JØB       0393,ANDRUS
            PRØGRAM 0A393
            CALL REGAME
            END
            END
NE BLANK CARD INSERTED HERE

  ØNE       SET ØF 4 INPUT CARDS



 ANØTHER SET ØF 4 INPUT CARDS IF DESIRED



  ETC.
```

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) U. S. Naval Postgraduate School, Monterey, California	2a. REPORT SECURITY CLASSIFICATION Unclassified	2b. GROUP x
3. REPORT TITLE A Computer Simulation for the Evaluation of Surface-to-Air Missile Systems in a Clear Environment		
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) Task Progress		
5. AUTHOR(S) (Last name, first name, initial) Andrus, Alvin F.		
6. REPORT DATE June 1966	7a. TOTAL NO. OF PAGES 95	7b. NO. OF REFS 0
8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S) TR 67	
b. PROJECT NO. 02551		
c. WEPTASK R520 00 010/551 1/F018 0501	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
d.		
10. AVAILABILITY/LIMITATION NOTICES Distribution of this document is unlimited.		
11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY CNO (OP-06C)	

13. ABSTRACT

A probabilistic event store computer simulation of the interactions between surface-to-air missile systems and aircraft in a non-jamming environment and over flat terrain is presented. The purpose of the model is to test the general disposition of missile areas and the associated missile system reaction times against an aircraft attack. A complete description of the model with the flow charts and CDC-FORTRAN-60 program listing is included.

DD FORM 1 JAN 64 1473

UNCLASSIFIED
Security Classification

KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
omputer Simulation						
onte-Carlo						
EGAME						
urface-to-Air Missile Systems						
imulation						

INSTRUCTIONS

ORIGINATING ACTIVITY: Enter the name and address he contractor, subcontractor, grantee, Department of De- se activity or other organization (*corporate author*) issuing report.

REPORT SECURITY CLASSIFICATION: Enter the over- security classification of the report. Indicate whether estricted Data" is included. Marking is to be in accord- e with appropriate security regulations.

GROUP: Automatic downgrading is specified in DoD Di- tive 5200.10 and Armed Forces Industrial Manual. Enter group number. Also, when applicable, show that optional kings have been used for Group 3 and Group 4 as author- d.

REPORT TITLE: Enter the complete report title in all ital letters. Titles in all cases should be unclassified. meaningful title cannot be selected without classifica- , show title classification in all capitals in parenthesis ediately following the title.

DESCRIPTIVE NOTES: If appropriate, enter the type of ort, e.g., interim, progress, summary, annual, or final. e the inclusive dates when a specific reporting period is ered.

AUTHOR(S): Enter the name(s) of author(s) as shown on n the report. Enter last name, first name, middle initial. ilitary, show rank and branch of service. The name of principal author is an absolute minimum requirement.

REPORT DATE: Enter the date of the report as day, th, year; or month, year. If more than one date appears the report, use date of publication.

TOTAL NUMBER OF PAGES: The total page count uld follow normal pagination procedures, i.e., enter the ber of pages containing information.

NUMBER OF REFERENCES: Enter the total number of erences cited in the report.

CONTRACT OR GRANT NUMBER: If appropriate, enter applicable number of the contract or grant under which report was written.

8c, & 8*d*. PROJECT NUMBER: Enter the appropriate itary department identification, such as project number, project number, system numbers, task number, etc.

ORIGINATOR'S REPORT NUMBER(S): Enter the offi- l report number by which the document will be identified l controlled by the originating activity. This number must unique to this report.

OTHER REPORT NUMBER(S): If the report has been igned any other report numbers (*either by the originator by the sponsor*), also enter this number(s).

AVAILABILITY/LIMITATION NOTICES: Enter any lim- tions on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through ________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through ________________."

(5) "All distribution of this report is controlled. Qual- ified DDC users shall request through ________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indi- cate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explana- tory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring (*pay- ing for*) the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical re- port. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the in- formation in the paragraph, represented as *(TS), (S), (C), or (U)*.

There is no limitation on the length of the abstract. How- ever, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identi- fiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical con- text. The assignment of links, roles, and weights is optional.

DISTRIBUTION LIST

Documents Department
General Library
University of California
Berkeley, California 94720

Lockheed-California Company
Centeral Library
Dept. 77-14, Bldg. 170, Plt. B-1
Burbank, California 91503

Naval Ordnance Test Station
China Lake, California
Attn: Technical Library

Serials Dept., Library
University of California, San Diego
La Jolla, California 92038

Aircraft Division
Douglas Aircraft Company, Inc.
3855 Lakewood Boulevard
Long Beach, California 90801
Attn: Technical Library

Librarian
Government Publications Room
University of California
Los Angeles, California 90024

Librarian
Numerical Analysis Research
University of California
405 Hilgard Avenue
Los Angeles, California 90024

Chief Scientist
Office of Naval Research
Branch Office
1030 East Green Street
Pasadena, California 91101

Commanding Officer and Director
U. S. Navy Electronics Lab. (Library)
San Diego, California 92152

General Dynamics/Convair
P.O. Box 1950
San Diego, California 92112
Attn: Engineering Library
Mail Zone 6-157

Ryan Aeronautical Company
Attn: Technical Information
Services
Lindbergh Field
San Diego, California 92112

General Electric Company
Technical Information Center
P.O. Drawer QQ
Santa Barbara, California 93102

Library
Boulder Laboratories
National Bureau of Standards
Boulder, Colorado 80302

Government Documents Division
University of Colorado Libraries
Boulder, Colorado 80304

The Library
United Aircraft Corporation
400 Main Street
East Hartford, Connecticut 06108

Documents Division
Yale University Library
New Haven, Connecticut 06520

Librarian
Bureau of Naval Weapons
Washington, D. C. 20360

George Washington University Library
2023 G Street, N. W.
Washington, D. C. 20006

National Bureau of Standards Library
Room 301, Northwest Building
Washington, D. C. 20234

Director
Naval Research Laboratory
Washington, D. C. 20390
Attn: Code 2027

University of Chicago Library
Serial Records Department
Chicago, Illinois 60637

Documents Department
Northwestern University Library
Evanston, Illinois 60201

The Technological Institute, Library
Northwestern University
Evanston, Illinois 60201

Librarian
Purdue University
Lafayette, Indiana 47907

Johns Hopkins University Library
Baltimore
Maryland 21218

Martin Company
Science-Technology Library
Mail 398
Baltimore, Maryland 21203

Scientific and Technical Information
Facility
Attn: NASA Representative
P.O. Box 5700
Bethesda, Maryland 20014

Documents Office
University of Maryland Library
College Park, Maryland 20742

The Johns Hopkins University
Applied Physics Laboratory
Silver Spring, Maryland
Attn: Document Librarian

Librarian
Technical Library, Code 245L
Building 39/3
Boston Naval Shipyard
Boston, Massachusetts 02129

Massachusetts Institute of Technology
Serials and Documents
Hayden Library
Cambridge, Massachusetts 02139

Technical Report Collection
303A, Pierce Hall
Harvard University
Cambridge, Massachusetts 02138
Attn: Mr. John A. Harrison, Librarian

Alumni Memorial Library
Lowell Technological Institute
Lowell, Massachusetts

Librarian
University of Michigan
Ann Arbor, Michigan 48104

Gifts and Exchange Division
Walter Library
University of Minnesota
Minneapolis, Minnesota 55455

Reference Department
John M. Olin Library
Washington University
6600 Millbrook Boulevard
St. Louis, Missouri 63130

Librarian
Forrestal Research Center
Princeton University
Princeton, New Jersey 08540

U. S. Naval Air Turbine Test Station
Attn: Foundational Research Coordinator
Trenton, New Jersey 08607

Engineering Library
Plant 25
Grumman Aircraft Engineering Corp.
Bethpage, L. I., New York 11714

Librarian
Fordham University
Bronx, New York 10458

U. S. Naval Applied Science Laboratory
Technical Library
Building 291, Code 9832
Naval Base
Brooklyn, New York 11251

Librarian
Cornell Aeronautical Laboratory
4455 Genesee Street
Buffalo, New York 14225

Central Serial Record Dept.
Cornell University Library
Ithaca, New York 14850

Columbia University Libraries
Documents Acquisitions
535 W. 114 Street
New York, New York 10027

Engineering Societies Library
345 East 47th Street
New York, New York 10017

Library-Serials Department
Rensselaer Polytechnic Institute
Troy, New York 12181

Librarian
Documents Division
Duke University
Durham, North Carolina 27706

Ohio State University Libraries
Serial Division
1858 Neil Avenue
Columbus, Ohio 43210

Commander
Philadelphia Naval Shipyard
Philadelphia, Pennsylvania 19112
Attn: Librarian, Code 249c

Steam Engineering Library
Westinghouse Electric Corporation
Lester Branch Postoffice
Philadelphia, Pennsylvania 19113

Hunt Library
Carnegie Institute of Technology
Pittsburgh, Pennsylvania 15213

Documents Division
Brown University Library
Providence, Rhode Island 02912

Central Research Library
Oak Ridge National Laboratory
Post Office Box X
Oak Ridge, Tennessee 37831

Documents Division
The Library
Texas A & M University
College Station, Texas 77843

Librarian
LTV Vought Aeronautics Division
P.O. Box 5907
Dallas, Texas 75222

Gifts and Exchange Section
Periodicals Department
University of Utah Libraries
Salt Lake City, Utah 84112

Defense Documentation Center (DDC)
Cameron Station
Alexandria, Virginia 22314
Attn: IRS (20 copies)

FOREIGN COUNTRIES

Engineering Library
Hawker Siddeley Engineering
Box 6001
Toronto International Airport
Ontario, Canada
Attn: Mrs. M. Newns, Librarian

Exchange Section
National Lending Library for
Science and Technology
Boston Spa
Yorkshire, England

The Librarian
Patent Office Library
25 Southampton Buildings
Chancery Lane
London W. C. 2., England

Librarian
National Inst. of Oceanography
Wormley, Godalming
Surrey, England

Dr. H. Tigerschiöld, Director
Library
Chalmers University of Technology
Gibraltargatan 5
Gothenburg S, Sweden

Chief of Naval Operations (OP-06C)
Room 5E531
The Pentagon
Washington, D. C. (5 copies)

Text Library
U. S. Naval Postgraduate School
Monterey, California (75 copies)

Department of Operations Analysis
U. S. Naval Postgraduate School
Monterey, California (20 copies)